PolyPublie
Polytechnique Montréal

POLYTECHNIQUE MONTRÉAL
UNIVERSITÉ D'INGÉNIERIE

Titre: Title:	Automatic Behavior Analysis and Understanding of Collision Processes Using Video Sensors
Auteur: Author:	Mohamed Gomaa M. Mohamed
Date:	2015
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Mohamed, M. G. M. (2015). Automatic Behavior Analysis and Understanding of Collision Processes Using Video Sensors [Thèse de doctorat, École Polytechnique de Montréal]. PolyPublie. https://publications.polymtl.ca/1784/

Document en libre accès dans PolyPublie
Open Access document in PolyPublie

URL de PolyPublie: PolyPublie URL:	https://publications.polymtl.ca/1784/
Directeurs de recherche: Advisors:	Nicolas Saunier
Programme: Program:	Génie civil

Ce fichier a été téléchargé à partir de PolyPublie, le dépôt institutionnel de Polytechnique Montréal
This file has been downloaded from PolyPublie, the institutional repository of Polytechnique Montréal

https://publications.polymtl.ca

UNIVERSITÉ DE MONTRÉAL

AUTOMATIC BEHAVIOR ANALYSIS AND UNDERSTANDING OF COLLISION PROCESSES USING VIDEO SENSORS

MOHAMED GOMAA M. MOHAMED
DÉPARTEMENT DES GÉNIES CIVIL, GÉOLOGIQUE ET DES MINES
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

THÈSE PRÉSENTÉE EN VUE DE L'OBTENTION
DU DIPLÔME DE PHILOSOPHIÆ DOCTOR
(GÉNIE CIVIL)
JUILLET 2015

© Mohamed Gomaa M. Mohamed, 2015.

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Cette thèse intitulée :

AUTOMATIC BEHAVIOR ANALYSIS AND UNDERSTANDING OF COLLISION PROCESSES USING VIDEO SENSORS

présentée par : MOHAMED Mohamed Gomaa M.
en vue de l'obtention du diplôme de : Philosophiae Doctor
a été dûment acceptée par le jury d'examen constitué de :

M. TRÉPANIER Martin, Ph. D., président
M. SAUNIER Nicolas, Ph. D., membre et directeur de recherche
M. MORRIS Brendan, Ph. D., membre
M. ALECSANDRU Ciprian, Ph. D., membre

DEDICATION

To my dear parents and my beloved wife and kids

ACKNOWLEDGMENTS

First and foremost, I would like to thank Allah for His endless blessings and for keeping my body and mind strong to fulfill this research. Thanks Allah for paving the way for coming to Montreal and continuing my study in Polytechnique Montreal without any intended plans.

I would like to express my sincere appreciation to my supervisor, Dr. Nicolas Saunier, for his invaluable contributions, constructive feedbacks and guidance through my research journey. This research would not have been possible without his guidance and support. Thank you for answering my questions quickly at any time and your careful revision of this thesis. Indeed, no words can fulfill the feeling of gratitude I carry: I was fortunate to work under your supervision.

I would like to acknowledge the financial support of the Natural Sciences and Research Council of Canada (NSERC). I would also like to thank Zu Kim of California PATH and Ann Stansel of the Kentucky Transportation Cabinet for providing the Kentucky video dataset, the City of Montreal for providing collision diagrams of the studied intersections in Montreal and financial support. I would like to thank Steve Warren (SNC-Lavalin O&M inc.) for helping us to collect the video data of the Guy Intersection.

I would like to thank my lab-mates for supporting me, in particular Kinan Bahbouh and to all my friends that supported and accompanied me all these years, in particular Ibrahim Bakry, Ibrahim Mashhour, and Adham Tamam.

I also thank my beloved parents for everything they have done in my life and for their moral support throughout my research. I am also grateful to my brother Osama for happily carrying loads of responsibilities that I burdened him with while away for study. I must acknowledge my younger sisters, Basma, Eman, and Nada for encouraging me with their best wishes.

At last, and most importantly, I am extremely thankful to my wife, Mona Abouhamad for living and interacting with me during the research journey with its ups and downs. She always supports me with her love and encouragement till completing the thesis. I want to express my deep love to my kids, Sara and Yusuf, for sharing our dream.

RÉSUMÉ

La sécurité routière est un des problèmes de société les plus importants à cause des multiples impacts et coûts des accidents de la route. Traditionnellement, le diagnostic de sécurité repose principalement sur les données historiques de collision. Cette approche réactive mène à remédier au problème de sécurité après que ses impacts sur la société soit déjà réalisés. Les analystes de la sécurité et les décideurs doivent attendre jusqu'à ce qu'un nombre suffisant de collisions (ce qui demande d'attendre habituellement au moins trois ans) soit collecté pour analyser ou mettre en place des mesures d'amélioration de la sécurité routière. Les méthodes substituts (« surrogate ») d'analyse de la sécurité constituent une approche alternative proactive qui s'appuie sur l'observation d'événements « dangereux » sans collision, souvent appelé accidents « évités de justesse » (« near misses ») ou « conflits ». Parmi ces approches, les techniques de conflits de trafic (TCT) reposent sur la collecte des données de conflit par des observateurs sur le terrain qui interprètent leur sévérité. Par conséquent, les TCT souffrent des variations de jugement des observateurs, de la difficulté de mesurer les indicateurs de sécurité en temps réel par les observateurs, et du coût de la collecte des données.

Cette thèse vise à proposer un cadre générique et efficace pour utiliser et gérer de grandes quantités de riches données spatio-temporelles obtenues par analyse vidéo automatique afin d'effectuer l'interprétation automatique de la scène, la compréhension du comportement des conducteurs, la détection des comportements anormaux et l'analyse substitut de la sécurité. À cette fin, ce cadre a deux composantes principales : 1) un cadre d'apprentissage multi-niveau des mouvements et 2) un cadre d'analyse substitut de la sécurité.

Pour commencer, un outil d'analyse vidéo venant du projet au code source ouvert « Traffic Intelligence » est utilisé. Un algorithme de lissage est développé pour les trajectoires extraites et sa performance est évaluée quantitativement. Les performances de l'algorithme sont évaluées dans trois études de cas. La mesure de performance est améliorée de 86-95 % pour tous les usagers de la route, de 97 % pour les véhicules, et seulement de 80-86 % pour les piétons.

Le cadre d'apprentissage multi-niveau des mouvements est utilisé pour l'interprétation de scène automatisée et la détection de comportements anormaux. Tout d'abord, les zones d'intérêt (zones d'entrée/sortie, zones d'occlusion, et zones de bruit résultant d'erreur de l'analyse vidéo) sont détectées par le biais d'un modèle de mélange de gaussiennes et l'algorithme de espérance-

maximisation. De plus, nous présentons quatre applications différentes des zones détectées : filtrage des trajectoires, connexion des trajectoires divisées, nettoyage de l'apprentissage des mouvements et de la prédiction des mouvements. Ces opérations reposent sur trois algorithmes non-supervisés; 1) un algorithme de filtrage, 2) un algorithme de connexion, et 3) un algorithme de nettoyage. Deuxièmement, les patrons de mouvement (PM), représentés par des trajectoires prototypes, sont appris à partir des trajectoires des usagers en deux phases, une première fondée sur la les informations spatiales puis une seconde sur les informations temporelles (de vitesse). Enfin, les trajectoires qui ne sont pas assignées à un PM après l'apprentissage sont considérées comme des anomalies. Une large étude expérimentale de trois cas variés démontre les capacités du cadre de plusieurs façons : elle réduit le coût de calcul d'environ 90 % pour l'apprentissage des mouvements, connecte les trajectoires incomplètes avec une précision jusqu'à 97 %, élimine les trajectoires aberrantes et permettent de prédire les mouvements futurs de façon réaliste pour calculer des indicateurs de sécurité.

Pour l'analyse substitut de la sécurité, trois questions différentes sont étudiés concernant : 1) l'influence de la méthode de prédiction du mouvement sur les indicateurs de sécurité, 2) l'influence de la représentation géométrique des usagers sur l'exactitude des indicateurs de sécurité, et 3) la façon d'interpréter les indicateurs de sécurité. Pour répondre à ces questions, l'approche est appliquée au diagnostic de la sécurité des interactions entre les véhicules tournant à gauche et les véhicules en direction opposée à un carrefour à feux.

Premièrement, la prédiction du mouvement est nécessaire pour déterminer si deux usagers de la route sont sur une trajectoire de collision et pour calculer des indicateurs de sécurité continus tels que le temps à la collision (TTC). La méthode par défaut, utilisée sans justification dans une grande partie de la littérature, est la prédiction à vecteur-vitesse constante. Toutefois, le cadre générique proposée prédit les positions futures des usagers de la route selon des différentes hypothèses d'extrapolation : des méthodes cinématiques (MC) telles que la prédiction à vecteur-vitesse constant et l'adaptation aléatoire basée sur un échantillonnage de distributions d'accélération et de direction, et les méthodes par appariement aux patrons de mouvement (APM). L'APM consiste à apparier les portions des trajectoires des usagers jusqu'à un instant donné avec les prototypes appris à l'étape de l'interprétation de scène. Les résultats montrent que l'APM est en mesure de calculer les indicateurs de sécurité dès le début de l'interaction et fournit un plus grand nombre de mesures dans le temps. Cela confirme que les MC ne sont pas réalistes pour calculer les indicateurs de la

sécurité pour les mouvements de virage et pour comprendre les processus de collision. L'intégration de l'interprétation de scène avec l'analyse de sécurité semble capable d'effectuer le diagnostic de sécurité d'une intersection dans un système automatisé et de manière non supervisée.

Deuxièmement, différentes représentations géométriques des usagers de la route sont proposées pour la mesure automatisée des indicateurs de sécurité. En particulier, ils sont utilisés pour automatiser les mesures du temps post-empiètement (PET). Les résultats montrent que la représentation d'un usager de la route comme une enveloppe convexe basée sur les points caractéristiques est la plus précise avec une erreur quadratique moyenne égale à 0,25 s.

Enfin, deux approches d'interprétation des indicateurs de sécurité sont étudiées pour l'analyse de la sécurité : 1) les distributions des indicateurs agrégés de sécurité et 2) la classification des profils temporels des indicateurs. La première approche repose sur la comparaison des distributions des indicateurs agrégés de sécurité. La seconde approche a pour objectif d'analyser l'ensemble des séries temporelles des indicateurs pour trouver des similarités entre les interactions, alors que la plupart des méthodes d'interprétation actuelles des indicateurs continues de sécurité (comme la première approche) reposent sur une seule valeur d'indicateur à un moment donné pour qualifier de l'ensemble de l'interaction, comme le TTC minimum. Une nouvelle mesure de similarité pour les séries temporelles, la plus longue sous-séquence commune alignée, est appliquée à des mesures substituts de sécurité et d'autres indicateurs caractérisant les interactions des usagers. La nouvelle mesure de similarité est couplée avec un algorithme personnalisé de regroupement qui ne nécessite pas de connaître le nombre de groupes attendu et reste interprétable grâce à l'utilisation de profiles d'indicateurs prototypes comme représentants de chaque groupe. Un résultat important est que la classification des indicateurs peut éliminer l'influence du bruit des mesures, qui pourrait mener à surestimer le danger d'une interaction. L'approche de classification des indicateurs aide à mieux comprendre les processus de collision dans leur ensemble.

ABSTRACT

Traffic safety is one of the most important social issues due to the multiple costs of collisions. Traditionally, safety diagnosis depends mainly on historical collision data. This reactive approach leads to remedy the existing safety problem after the materialization of the induced social cost. Safety analysts and decision makers must wait till a sufficient number of collisions (typically at least 3 years of collision data) is collected to analyze and to devise countermeasures. Surrogate safety analysis is an alternative and proactive approach that relies on the observation of traffic events without a collision, in particular “unsafe” events often called “near misses” or “conflicts”. Among these approaches, traffic conflict techniques (TCT) rely mainly on field observers to identify conflicts and interpret their severity. Consequently, TCTs suffer from the variations of observer judgement, the cost of collecting conflict data, and the difficulty of measuring safety indicators in real time by the observers.

This dissertation aims to propose an effective and generic framework to utilize and manage the rich amount of spatial-temporal data obtained through automated video analysis for automated scene interpretation, driver behaviour understanding, anomalous behaviour detection, and surrogate safety analysis. To this end, this framework is composed of two main components: 1) a multi-level motion pattern learning sub-framework and 2) a surrogate safety analysis sub-framework.

At the beginning, a video analysis tool from the open source project “Traffic Intelligence” is used. A smoothing algorithm is developed for the extracted trajectories and its performance is evaluated quantitatively. The performance of the algorithm is evaluated for three case studies. The performance measure is improved by 86-95 % for all road users, up to 97 % for vehicles, and only up to 80-86 % for pedestrians.

The proposed multi-level motion pattern learning framework is used for automated scene interpretation and anomalous behaviour detection. Firstly, the interest zones (Entry/Exit zones, occluded zones, and noise zones due to tracking failure) are detected through a Gaussian Mixture Model and the Expectation Maximization algorithm. Meanwhile, we introduce four different applications of the detected zones: filtering trajectories, connecting divided trajectories, cleaning the dataset, and speeding up motion pattern learning and prediction methods. This is performed with three unsupervised algorithms; 1) a filtering algorithm, 2) a connection algorithm, and 3) a cleaning algorithm. Secondly, motion patterns, represented by trajectory prototypes, are learnt from

road users' trajectories using two-stage trajectory learning based on spatial then temporal (speed) information. Finally, the trajectories that are not assigned to any MP after the learning are considered as anomalies. A comprehensive experimental analysis of three varied scenes demonstrates the framework ability in several ways: it reduces the computation cost up to 90 % for motion pattern learning, connects incomplete trajectories with performance accuracy up to 97 %, removes outliers from trajectory dataset, and predicts future motion realistically to compute safety indicators.

For surrogate safety analysis, three different questions are investigated: 1) what is the influence of motion prediction on the measurement of safety indicators, 2) what is the influence of the road users' geometric representation on the accuracy of estimated safety indicators, 3) how can the safety indicators be interpreted. To answer these questions, the framework is applied to the safety diagnosis of left turn and opposite direction interactions at a signalized intersection.

First, motion prediction is required to identify whether two road users are on a collision course and to compute several continuous safety indicators such as Time to Collision (TTC). The default, unjustified method used in much of the literature is prediction at constant velocity. However, the proposed generic framework predicts road users' future positions depending on different extrapolation hypotheses: kinematic methods (KM) such as constant velocity and random adaptation based on sampling distributions of acceleration and direction, and motion pattern matching (MPM). MPM consists in matching partial road user trajectories up to the current instant to the learnt prototypes obtained from the scene interpretation step. The results show that MPM is able to compute the safety indicators early with a larger number of measurements. This finding supports that KM are not realistic enough to compute safety indicators for turning movements and to understand collision processes. The integration of the scene interpretation with the safety analysis is able to perform the safety diagnosis of an intersection in an automated and unsupervised fashion.

Second, different geometric representations of road users are proposed for automated measurement of safety indicators. In particular, they are compared in the case of the measurements of post encroachment time (PET). The results show that representing the road user as a convex hull using extracted features is the most accurate with a root mean square error equal to 0.25 s for PET computation.

Finally, two interpretation approaches of safety indicators are investigated for safety analysis: 1) aggregated safety indicator distributions and 2) indicator profile classification. The first approach relies on the comparison of the distributions of aggregated safety indicators. The second approach aims to compare the whole time series or profiles of the indicators to find similarities between interactions, while most current interpretations (such as the first approach) of continuous safety indicators rely on only one indicator value at a given time to qualify the whole interaction, e.g. the minimum TTC. A new similarity measure for time series, the aligned longest common sub-sequence, is applied to surrogate measures of safety and other indicators characterizing road user interactions. The new similarity measure is paired with a custom clustering algorithm that does not require to set the number of expected clusters and remains interpretable through the use of prototype indicator profiles as cluster representatives. An important finding is that classifying indicators can eliminate the influence of noisy measurements, which could lead to over-estimate the risk for such interactions. The classification approach helps to better understand collision processes.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF INITIALS AND ABBREVIATIONS

ALCSS	Aligned Longest Common Sub-Sequence
AP	Activity Path
AR	Attainable Region
BB	Bounding Box
BIC	Bayesian Information Criteria
CBM	Centroid Based Method
CirBM	Circle Based Method
CV	Constant Velocity
CP	Collision Point
CSJ	Cumulative Squared Jerk
CZ	Crossing Zone
DP	Dirichlet Processes
DTMD	Directional Trimmed Mean Distance
DTW	Dynamic Time Warping
EM	Expectation Maximization
EVT	Extreme Value Theory
FBCH	Features-Based using Convex Hull method
FHWA	Federal Highway Administration
FMAB	Feature-based using Minimum Area Bounding box method
FOV	Field of View
FCM	Fuzzy C Mean
GMM	Gaussian Mixture Model
GP	Gaussian Processes
GPD	Generalized Pareto Distribution
KLT	Kanada- Lucas- Tomasi
HCA	Hierarchical Clustering Analysis
HDP	Hierarchical Dirichlet Process
HMM	Hidden Markov Model
ICSTCT	International Calibration Study of Traffic Conflict Techniques
LCSS	Longest Common Sub-Sequence

LDA	Latent Dirichlet Allocation
LT	Left Turn
LTOD	Left Turn and Opposite Direction
MP	Motion Pattern
MPCS	Motion Pattern Constant Speed
MPM	Motion Pattern Matching
MPSP	Motion Pattern Speed Profile
NA	Normal Adaptation
OD	Opposite Direction
PCA	Principle Component Analysis
pdf	probability density function
PDO	Property Damage Only
PET	Post Encroachment Time
pLSA	probabilistic Latent Semantic Analysis
pPET	predicted Post Encroachment Time
POI	Points of Interest
PS	Set of Initial Positions
SAAQ	Société de l'Assurance Automobile du Québec (Quebec's auto insurance company)
SSAM	Surrogate Safety Assessment Model
TA	Time to Accident
TCT	Traffic Conflict Technique
TET	Time Exposed TTC
TI	Traffic Intelligence
TIT	Time Integrated TTC
TTC	Time To Collision
WHO	World Health Organization

GLOSSARY OF TERMS

Collisions	A collision is defined as an interaction where two or more vehicles enter in contact or one vehicle goes into an obstacle or off the road. It leads to different scales of severity. Collisions are the basis for traditional road safety analysis.
Collision course	For a given motion prediction method, a collision course is defined as a situation in which two road users would collide if they follow the predicted trajectories.
Collision severity	It refers to the level of injury or property damage of single or multiple road user collisions.
Encounter	An encounter is an interaction meeting some criteria intended to select a subset of interactions that are "closer" to collisions, for which safety indicators such as TTC may be computed, or are below some threshold.
Interaction	An interaction is defined as a traffic event in which two road users exist simultaneously and are close enough.
Motion Pattern	The motion patterns are representations of the frequent road user movements in the studied site.
Post Encroachment Time (PET)	PET is measured directly using the actual trajectories: it is defined as the time difference between the instant when the first road user leaves the crossing zone and the instant when the second road user enters it.
Points of interest (POIs)	The POIs indicate regions where road users enter, exit, and stop. They are learnt from the origins and destinations of observed trajectories. In this thesis, POIs correspond mainly to the entry/exit zones.
Road user spatial information	The sequence of positions in a trajectory.

Road user temporal information	The information derived from a trajectory by differentiation such as speed and acceleration. In this thesis, temporal information is limited to speed.
Safety Indicator	A safety indicator is an operational parameter that is used to measure the proximity to a potential collision, or probability of collision, or the severity of the potential collision. It is synonymous with severity indicator.
Time to collision (TTC)	It is defined for a given motion prediction method as the time required for two vehicles to collide following the predicted trajectories.
Trajectory	It is defined as a sequence of positions, typically one position for each frame of the video.

LIST OF APPENDICES

CHAPTER 1 INTRODUCTION

1.1 Context

Traffic safety is one of the most important social issues due to the multiple costs of collisions. The total social cost of road collision in Canada in 2004 was estimated at $62.7 billion annually (about 4.9 % of Canada's 2004 Gross Domestic Product) (Vodden, Smith, Eaton, & Mayhew, 2007). In Quebec, the Quebec's auto insurance company (SAAQ – Société de l'Assurance Automobile du Québec) reported in the recent road collision report that 336 people were killed and 1573 were sever injured in road collisions in 2014 (SAAQ, 2015). Globally, the World Health Organization (WHO) ranked road accidents in the ninth place of leading causes of death and disability in 2009 and predicted it will rise to the fifth place by 2030 (WHO, 2009). The 2013 WHO report confirms the trend is going worse and road accidents are ranked in the eighth leading cause of death which signifies an urgent action must be taken (WHO, 2013). Traffic safety diagnosis relies on two main approaches (see Figure 1.1): collision –based analysis and surrogate safety analysis. Each of these approaches has its own advantages and disadvantages.

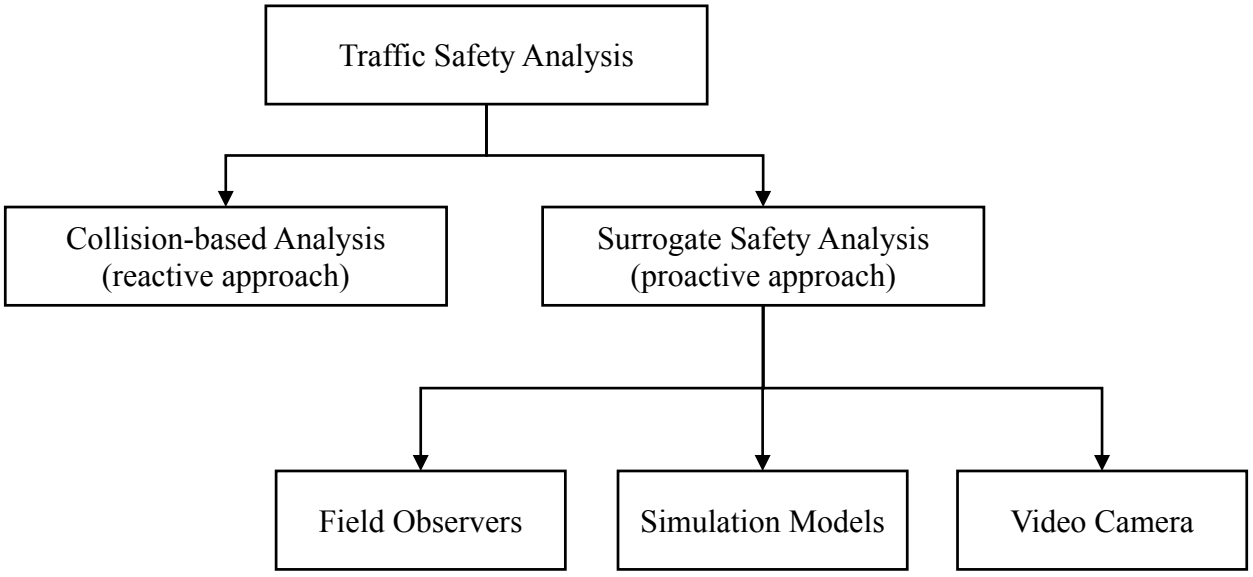


Figure 1.1: Traffic safety analysis main approaches

Safety manuals such as the manual of the World Road Association (RSM, 2003) depend mainly on historical collision data obtained from police and hospital reports and on different types of statistical analysis to identify and understand the failures of road systems, and to propose corrective

actions. This type of data has several shortcomings: the underreporting of some types of accidents, the lack of information in the reports, and the relatively small number of events. Besides, the record is done after the accident happened. The safety analysts and decision makers must wait till a sufficient number of accidents is collected to analyze the collisions and to devise countermeasures. The typical period of collision data is at least three years for safety analysis (Nicholson, 1985). Therefore, collision-based safety analysis is a reactive approach and the existing safety problem may only be remedied after the materialization of the induced social cost. These limitations have lead researchers to search for new methods to perform road safety diagnosis with higher confidence and in a proactive manner. This recommendation is contrasted with our first attempt to understand collision factors based on traditional collision data (Mohamed M. G., Saunier, Miranda-Moreno, & Ukkusuri, 2013).

One promising approach relies on the observation of traffic events without a collision, in particular "unsafe" events often called "near misses" or "conflicts". Such methods have been developed in several countries since the late 1960s and are now better known as methods for surrogate safety analysis. One commonly used, and probably the first invented, category of surrogate safety measurement tools is the Traffic Conflict Technique (TCT) which relies on the observation of conflicts. The widely accepted definition of a conflict according to the International Calibration Study of Traffic Conflict Techniques (ICSTCT) is as follows:

> *"A conflict is an observable situation in which two or more road users approach each other in time and space for such an extent that there is a risk of collision if their movements remain unchanged".* (Hydén & Amundsen, 1977)

The traditional TCTs rely mainly on human observers in the field to identify conflicts and interpret their severity. TCT thus suffers from intra-observer and inter-observer variations, in additional to the cost of conducting traffic conflict collections. The intra-observer variations and consistency problem are related to the variance of the observations made by the same observer when evaluating traffic events. The inter-observer variation concerns the potentially contradicting assessment of the same event by different observers. Operational indicators are used to measure the proximity to a potential collision, or probability of collision, and in some TCTs the severity of the potential collision. The term "severity" is typically used as an indication of the consequence of a collision in terms of property damage or injuries of one or more road users. Since a potential or hypothetical

collision is studied, its consequence is difficult to determine. Therefore, the general term safety indicator is used in this thesis to measure the safety of a traffic event to avoid confusion (see section 2.4.2 for more details). The safety indicator values are mostly calculated at a given time using the positions and speeds of the two interacting road users which cannot be measured in real time by human observers.

The development of microscopic simulation models provides a unique opportunity to conduct various advanced level analyses in the transportation engineering field. Among others, the ability to test possible strategies before their implementation is an important advantage of simulation. The Surrogate Safety Assessment Model (SSAM) was designed by the Federal Highway Administration (FHWA) to perform the analysis of trajectory data extracted from microscopic simulation software (e.g. VISSIM) (Gettman & Head, 2003; Gettman, Pu, Sayed, & Shelby, 2008). SSAM can calculate several safety indicators and the total number of conflicts. The traffic simulation models have several limitations such as the difficult calibration and validation of the models, the difficulty to represent driver behaviour well in particularly complex situations, and the perfect driver behaviour models that avoid all collisions. A detailed safety assessment using microscopic simulation model analysis is conducted by (Archer, 2005). The author concluded that existing simulations models do not yet represent sufficiently realistic road user behaviour (e.g. speed profiles in intersections and/or behaviour interactions with other users).

A possibility to reduce the limitations of surrogate measures of safety collected by field observers and simulation models is to collect automatically microscopic data (trajectories, i.e. series of positions), in particular using video sensors and computer vision techniques. A trajectory is defined as a sequence of positions, typically one position for each frame of the video. Hereafter, automated safety analysis based on video data can be conducted. This approach provides detailed microscopic data and offers a complementary approach to address the issues of manual collection methods. Yet, this automated approach is still under development and is not widely used.

For surrogate safety analysis, a key defining concept of all safety-relevant traffic events, is the collision course, i.e. a situation in which two road users would collide if their movements remain unchanged (taken from the conflict definition in (Hydén & Amundsen, 1977)). This requires specifying a method to predict the road users' movements in order to evaluate whether they are on a collision course, and to measure several safety indicators such as the Time to Collision (TTC).

TTC is defined for a given motion prediction method as the time required for two vehicles to collide following the predicted trajectories. Most analyses rely on the rarely specified or justified method of extrapolation at constant velocity, while several possible paths may in general lead road users to collide. This uncertainty in motion prediction is the result of the following factors:

- unobserved variables, e.g. the characteristics of the driver and the vehicle (if any), including the driving skills and ability to perform an evasive action, the awareness of road users to each other and their environment;
- the stochastic nature of predicting the future given the current state of the system, e.g. the variability of motion choices (small variations in speed and direction), the complexity of all the road users' interactions[1];
- measurement uncertainty, depending on the accuracy of the sensing technology.

The choice of a method for motion prediction is essential for surrogate safety analysis. Such methods are very similar to navigation and path planning in robotics, where collisions should be predicted and avoided. The difference is that robots know their goals, in particular places to reach, and can plan accordingly, while the analysis of road user interactions based on exterior observations does generally not have access to their internal state and goals. Therefore, a possible assumption is to iterate the future positions based on the assumed driver intention as a control input (e.g. acceleration and steering). Although this assumption considers partially driver behaviour, it does not consider the traffic environment (e.g. intersection layout, curbs, sidewalks locations). This shortcoming is particularly visible for turning vehicles. An appropriate solution is to learn the frequent road user motions, known as Motion Patterns (MPs), on the studied site. These patterns help to interpret the scene and to understand the road users' behaviour, and they can be used for motion prediction in a probabilistic manner to compute several safety indicators. The automated learning of MPs is a challenging problem in the case of a large number of trajectories. New algorithms are therefore needed to mine these massive datasets while having a manageable computational cost in time and space. Moreover, the characteristics of the behaviour of road users

[1] In this thesis, an interaction is defined as a traffic event in which two road users exist simultaneously and are close enough. Events involving only one road user are not considered in this work.

can be described in terms of both trajectories (spatial information[2]) and speed profiles (temporal information) with respect to the traffic environment. However, limited work is documented to cluster (or even consider) individual speed profiles, separately from motion patterns. The joint or hierarchical learning of MPs and speed profiles and their applications for behavior analysis should be investigated.

1.2 Challenges

A generic framework is required to compute reliable surrogate safety measures. Therefore, four main topics have been studied in this research. Each topic has its own challenges as summarized in Table 1.1. Addressing these challenges and incorporating them will provide a robust approach for surrogate safety analysis.

Table 1.1: Current challenges of different topics for improving safety analysis approach

Topic	**Current challenges and limitations**
Computer vision	• Varying quality of extracted trajectories e.g. with presence of noise.
Motion prediction models	• Most analyses are based on constant velocity model. • In robotics field, motion prediction for collision avoidance is a well-researched topic. These methods have not been investigated for surrogate safety analysis. • There is limited use of motion pattern methods for surrogate safety analysis. Few models have been proposed that include the variations of the speed profiles in motion prediction methods, for example for turning vehicles at an intersection.

[2] Spatial information refers to the set of positions in a trajectory

Table 1.1: Current challenges of different topics for improving safety analysis approach (cont'd)

Topic	Current challenges and limitations
Scene interpretation analysis	• High computational cost in time and required storage space. • Unsupervised scene interpretation is a challenging problem for traffic scenes. It should provide a description of road user behaviour with respect to traffic rules and environmental structure, with or without prior knowledge of the scene. • Scene is interpreted by motion patterns learnt based on spatial information only, limited work is documented to learn individual speed profiles, separately from motion patterns.
Surrogate safety analysis	• There is no generic framework that incorporates the previous topics after addressing their limitations to compute reliable surrogate measures of safety. • Current automatic video-based methods for surrogate safety analysis consider road users as points or circles to simplify the computation. • The interpretation of continuous safety indicators over time is based only on one value and not on the whole time series.

1.3 Objectives

The main objective of this research is to develop a consistent and generic framework for automated scene interpretation, driver behavior understanding, anomalous behaviour detection, and surrogate safety analysis. To achieve the main objective, specific objectives are identified as follows:

- Collect traffic video data and extract the trajectories for all road users using computer vision.

- Enhance the trajectories extracted by an existing video tracking tool to improve behaviour analysis accuracy (e.g. trajectory smoothing).
- Understand and interpret scene activities by developing a multi-level motion pattern learning framework. This framework must be able to detect anomalous events and to predict future behaviour.
- Propose new motion prediction methods and compare them to measure the safety of road users' interactions.
- Study the influence of geometric representation of road user on the automated measure of safety indicators.
- Understand interaction processes by analyzing the similarities between indicator profiles over time.

1.4 Methodology Overview

To address the research objectives, the overview of the research framework is presented in Figure 1.2. The methodology conducted consists of two main phases:

a. In the first phase, road user trajectories are extracted from video data recorded with a fixed camera using a custom feature-based video tracker available in the open source "Traffic Intelligence" project (Saunier N. , 2014). An algorithm is developed to smooth the extracted trajectories using the feature trajectories dataset. An unsupervised multi-level motion pattern (MP) learning sub-framework is developed that is used for automated scene interpretation, driver behavior understanding, anomalous behaviour detection, and motion prediction application. Moreover, logical constraints are proposed to accelerate the processing time for learning MPs and predicting future road user motion.

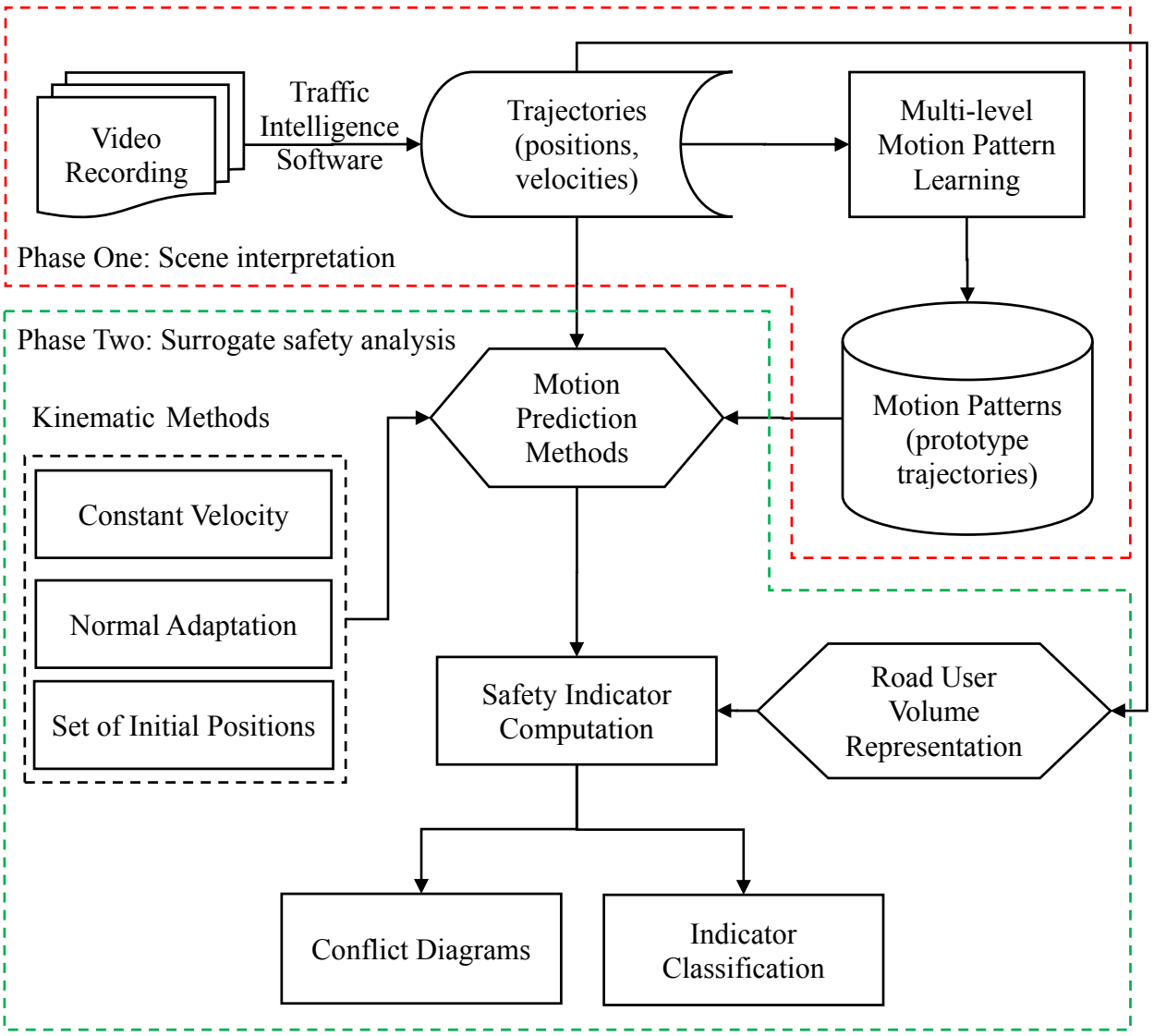


Figure 1.2: Research Overview

b. In the second phase, an automated safety diagnosis sub-framework is proposed and used mainly to analyse the Left Turn and Opposite Direction (LTOD) interactions at a signalized intersection. In the proposed sub-framework, different geometric representations of road user volume are proposed to compute safety indicators and evaluated to compute the Post Encroachment Time (PET) indicator. For the safety indicators that rely on motion prediction method such as TTC, different motion prediction methods, including the output of the MP learning framework, are investigated to evaluate whether two road users are on a collision course and to compute several safety indicators. These methods are grouped into two main categories: kinematic and motion pattern matching (MPM) methods. Kinematic

methods predict future motion based on a mathematical model using current vehicle state (e.g. current position and velocity). In this thesis, three methods are used: constant velocity, normal adaptation using control distribution function (e.g. acceleration and steering distribution) and set of initial position. On the other hand, MPM methods use the learnt prototypes, as prior knowledge of the studied site, to predict future motion according to the static traffic environment. Two safety indicators, TTC and a predicted version of PET (pPET), are calculated to assess the safety for the studied sites. The safety analysis is conducted by analyzing the distributions of indicators and by classifying similar indicator time series.

1.5 Contributions

The following contributions are done in this thesis:

1. An algorithm to reconstruct smooth road user trajectories based on feature trajectories.
2. The exploitation of points of interest[3] (POI), entry and exit zones, to discover static occlusion zones, filter trajectories, connect divided trajectories, detect outliers, and speed up MP learning and motion prediction.
3. MPs are extracted in two stages using spatial and temporal information. The integration of speed profiles in MPs is especially useful for motion prediction in the case of turning vehicles.
4. The application of the learnt prototypes (spatial and temporal) to anomaly detection, motion prediction and surrogate safety measures.
5. An investigation of different motion prediction methods to evaluate whether two road users are on a collision course and to compute safety indicators. Motion prediction based on following MPs proves to predict future behaviour realistically.
6. An investigation of different geometric representations of road user volume and their impact on the computation of safety indicators. It shows that the consideration of road

[3] The POIs indicate regions where road users enter, exit, and stop, learnt from the origins and destinations of tracked trajectories (see section 2.2.1).

users' outlines gives a more accurate estimation of PET in the case of an almost overhead camera view, while simple centroid based methods are more applicable for a low angle camera view.

7. Propose a probabilistic version of predicted post-encroachment time (pPET) indictor[4].
8. An open source software implementation (Mohamed M. G., 2014) and (Saunier N. , 2014) of the proposed methods to encourage adoption and further development.

1.6 Thesis Outlines

The remainder of this thesis is organized as follows:

Chapter 2 summarizes the literature review and identifies research gaps. The development of video analysis technology is presented and its application in the transportation domain. Additionally, the fields of behaviour analysis and scene interpretation are explained. Safety analysis methodologies are explored. Furthermore, the literature in the robotics field is reviewed to present the different methods for motion prediction and collision avoidance. Finally, the existing gaps and limitations are identified.

Chapter 3 introduces the methodology of the first phase of this research. The Traffic Intelligence project is explained. A novel smoothing algorithm is presented and a quantitative measure of the algorithm performance is discussed. A multi-level MP learning sub-framework is introduced for scene interpretation and anomaly detection.

Chapter 4 presents the application of the computer vision technique and the multi-level MP learning sub-framework on a comprehensive experimental analysis of three different intersections.

Chapter 5 presents the second phase of the methodology for surrogate safety analysis. Different motion prediction and several safety indicators are explained. The automated measurement of PET is explained using four different road users' representation methods. Two interpretation methods for continuous safety indicators are described.

[4] The implementation and the proposed methodology to compute pPET is a joint contribution with Nicolas Saunier. This was published in (Mohamed & Saunier, 2013a).

Chapter 6 presents the experimental safety analysis mainly of LTOD interactions in a signalized intersection. The influence of geometric representation of road user and different motion prediction methods on the measurement of safety indicators is discussed. The analysis of indicator distributions based on temporal aggregation and the analysis of the whole time series is investigated at the end to assess the safety of the interactions.

Chapter 7 concludes the thesis and discusses future improvements and applications.

CHAPTER 2 LITERATURE REVIEW

This chapter provides a review of the literature of related works. We start with a discussion of video tracking technology and the applications to transportation. Scene interpretation, an important application of video analysis and a core concept of the probabilistic safety analysis framework, is described showing the advantages and disadvantages of existing techniques. Additionally, we provide a discussion on previous development of safety analysis (traditional and surrogates approaches). Finally, a literature review of the research on motion prediction methods, in particular from the field of robotics, is presented.

2.1 Video Tracking Technology

Video tracking is the process of detecting and tracking objects of interest over time using video sensors. It has attracted lots of attentions because of rapid improvement of quality and resolution of video sensors (cameras), in addition to the improvement in computer power. Video sensors can be mounted at relatively high positions (e.g. light poles) on a selected road element. They can collect traffic data with several advantages over traditional spot detector such as radar, ultrasonic sensors, and inductive loop detectors. Video sensors are relatively easy to install, low-cost and can cover a large field of view. Moreover, they have the ability to capture naturalistic data with limited risk of road users changing their behaviour if they felt being watched. Using video tracking tools, rich and detailed information about all the road users can be extracted from video data such as trajectories, velocities, accelerations, lane changes, etc. This information can provide a high level of interpretation about individual road user behaviour and interaction behaviour with other road users. On the other hand, traditional sensors provide traffic data only at the location of the sensors. For example, loop detectors are embedded in pavements: hence, the installation and maintenance of these detectors require an interruption of traffic. Consequently, they are an expensive tool and difficult to maintain, so that a large number of sensors is often out of service.

Video analysis tools have been utilized in the field of transportation, in particular for the collection of road user positional data. This data is used in several applications such as counting road users, automated safety analysis, behaviour analysis, calibration and validation of microscopic simulation models. For instance, video analysis tools are developed by research teams in the following universities: Lund University (Laureshyn A. , 2010), University of Washington (Malinovskiy, Wu,

& Wang, 2009), University of Minnesota (Atev, Arumugam, Masoud, Janardan, & Papanikolopoulos, 2005), University of British Columbia (Saunier & Sayed, 2006; Ismail, 2010) and Polytechnique Montreal (Saunier N. , 2014).The latter one is available in an open source project with several tools for transportation data analysis, including a video analysis tool. This project, called "Traffic Intelligence (TI)", will be used in this research (for detail discussion about the video analysis tool available in TI, readers refer to Chapter 3). Methods for video analysis are classified into three main categories according to the level of automation: manual, semi-automated (supervised), and automated (unsupervised) (Maggio & Cavallaro, 2011)

- **Manual tracking:** The user detects objects of interest in each frame manually by watching the video film on a computer and annotating the objects by 'clicking' on them with a mouse. Objects of interest are accurately detected but it is very time consuming, and cannot be used for large volumes of video data. This method may be used in a portion of video clip to construct ground truth data for validation and performance calculation purposes. An example of an annotation tool developed at Polytechnique Montreal is available in the video tracking project called Urban Tracker (Jodoin, Bilodeau, & Saunier, 2014).

- **Automated tracking:** uses prior information about objects of interest that is defined mainly as a manual configuration input. Fully automated video-tracking technique is still under development in the field of computer vision and not widely used in real-world applications. These algorithms interpret pixels in an image sequence. Buch, Velastin, and Orwell (2011) reviewed the development of computer vision techniques for urban traffic analysis. Computer vision technique can overcome several limitations of field observers and manual video analysis such as the time-consuming and error-prone procedure.

- **Semi-automated tracking:** is a combination of manual and automated video trackers. It aims to employ computer aid in the processing of video records with an intervention of a user to obtain satisfactory results. In some cases, trained observers review a portion or all video clips manually for correcting and validating the quality of the tracked objects.

Video tracking consists of three main steps: defining an object of interest, object detection, and object tracking.

1. **The Object of interest** is defined based on the application. In transportation applications, objects of interest usually refer to the road users (vehicles, pedestrians and cyclists), while in building surveillance, they may refer to people and their behaviour.

2. **Object detection algorithm**, known also as object segmentation, aims to detect and identify the objects of interest in an image or video sequence. It is the foundation of the next process (i.e. object tracking) because only these objects will be considered later. The detection methods rely on either temporal or spatial information in the image sequence. The common approaches used for object detection are outlined as follows (Hu, Tan, Wang, & Maybank, 2004; Liu, Wu, & Zhang, 2008):

    - *Optical flow approaches* are methods to analyze relative motion between the objects and the scene in the sequence of images. The optical flow is the distribution of the apparent velocities of objects in an image. This approach relies on the characteristics of flow vectors of moving objects over time which detects moving objects in the image sequence. The main drawbacks of these approaches are the complex computation methods and sensitivity to both of noise and variations in scene illumination (Liu, Wu, & Zhang, 2008).

    - *Temporal differencing approaches* detect the moving region based on pixel wise difference between two or three consecutive frames in an image sequence. In (Lipton, Fujiyoshi, & Patil, 1998), an absolute difference of intensities in consecutive frames is computed, and then objects are extracted based on a defined threshold (e.g. 15 % of digitizer brightness range). The objects are clustered into motion regions using a connected component criterion. These approaches have low computational complexity. Despite the simplicity of these approaches, they are very sensitive to the selected threshold. A small threshold generates a lot of noise, however, a large one leads to miss a lot of information. In addition, the speed of objects affects the detection where speedy object can be treated as two objects (Yi & Liangzhong, 2010). Although these approaches are very adaptive to dynamic environments, they failed to detect all relevant features and the true shape of moving object (Spagnolo, Leo, D'Orazio, & Distante, 2004).

- *Background subtraction* is a common and effective method for object segmentation. Many background subtraction algorithms are built to detect moving object in an image, following almost the same scheme (Sobral & Vacavant, 2014):

  1) *Background initialization*: its goal is to build a reference background model,

  2) *Foreground detection*: its goal is to compare the current image with the background model, and

  3) *Background updating*: its goal is to update the initial background model during the detection process.

  There are several methods for background subtraction such as basic methods (e.g. absolute difference, mean, and variance), fuzzy methods, statistical methods using one or multiple Gaussians, and Eigen composition methods. Sobral (2013) offers a library of 37 background subtraction algorithms.

3. **Object *tracking algorithms*** can be classified into four major categories of algorithms (see (Hu, Tan, Wang, & Maybank, 2004; Saunier & Sayed, 2006; Liu, Wu, & Zhang, 2008) for more details). Each algorithm could be used alone or integrated with other algorithm to track objects. Each category is described as follows:

   - *Region based tracking algorithms* rely on identifying connected regions of the image (blobs) associated with each object, often obtained through background subtraction. Region-based algorithms are suitable in the free-flowing traffic condition. Congested traffic is considered as a serious problem for this category of algorithms. It groups more than one object in one large blob which leads to undercounting objects (Malik & Russell, 1997; Saunier & Sayed, 2006).

   - *Active contour based algorithms***:** a contour is commonly known as a "snake". The idea is to detect and track the moving object outline and keep updating it dynamically. Contour based algorithms are better than region based algorithms as they reduce the computational complexity (Malik & Russell, 1997). Nevertheless the problems stated previously remain.

   - *Model-based tracking algorithms* are performed in the following sequence steps (Koller, Daniilidis, Thorhallson, & Nagel, 1992). First, the algorithm needs to

choose an accurate wire-frame 3D model of typical objects. Second, these models are projected into the image plane. Finally, the projected models are matched with the detected objects and tracked in subsequence frames. While this algorithm achieve accurate trajectories, detailed information on object geometry is very difficult to obtain.

- *Feature-based tracking algorithms* perform the tracking and recognition of objects by extracting elements such as points instead of tracking them as a whole region. The algorithm can deal with the presence of partial occlusion by tracking visible parts of any object (Liu, Wu, & Zhang, 2008). This advantage makes the use of this algorithm possible in different lighting conditions: daylight, twilight, night-time and different traffic conditions (Saunier & Sayed, 2006). The Kanade-Lucas-Tomasi feature trackers (Tomasi & Kanade, 1991) are mostly used to track the features. The object consists at this stage of multiple features. The next procedure is grouping or clustering of these features to generate the object hypothesis. Feature-based tracking is the method of choice in the TI project and will be used in this thesis.

Nonetheless, a number of video analysis methods combine different tracking algorithms. For example, Jodoin et al (2014) combined the region based and feature based tracking algorithms. Region-based algorithm are used to estimate the object size and to group features; a feature-based algorithm is used to deal with the partial occlusion issue.

Video analysis can be used for traffic management to collect different traffic parameters such as vehicle counts and speeds which can also be classified. Bas, Tekalp and Salman (2007) counted vehicles from video using adaptive blob size for detecting and tracking vehicles. The adaptive blob method considers that the vehicles far away from the camera will appear as smaller blobs. Hence, this method estimates the relative distance between the camera and blob, then it adopts the blob size accordingly. Zhao and Wang (2013) proposed a new approach for counting vehicles through learning first semantic regions. The semantic regions classify and form the typical movement paths in the scene which facilitate the counting of vehicle in each path separately. Morris and Trivedi (2008b) used video analysis for two different traffic system: 1) the VEhicle Classifier and Traffic flOw analyzeR (VECTOR) module which is used for vehicle classification into eight different

types on highways, collect traffic statistic modelling (e.g. speed, flow, lane density), 2) a Path behaviour analysis module which deals with behaviour analysis and path modelling. In (Messelodi, Modena, & Zanin, 2005), the authors utilized a robust background and feature-based tracking method to track moving objects. The trajectories are used to estimate the speed profile of each object and to classify objects into seven categories based on model matching technique.

Automatic behaviour analysis is an important application and a very challenging problem. In the literature, machine learning and video analysis are integrated to understand activities in the scene. Existing techniques are discussed in detail in the next section.

One of the most challenging and useful application of video analysis is automated safety analysis. Atev et al (2005) proposed a vision based system to monitor intersection and predict a possible collision. The system can issue a warning message if a possible collision is predicted. Saunier and Sayed (2007) developed a system to extract traffic conflict automatically and analyzed the safety of a site using video sensors data. Zaki, Sayed, Ismail, and Alrukaibi (2012) identified traffic violations (spatial and temporal) by pedestrians using computer vision and machine learning techniques in two intersection cases. The authors defined the spatial violation as a pedestrian crossing an intersection through undesignated locations while temporal violation is defined if a pedestrian crossed an intersection during an improper signal phase. Jackson, Miranda-Moreno, St-Aubin, and Saunier (2013) presented the development of a mobile video camera system and of the open source TI project to collect and analyze video data. Three case studies are illustrated to show the effectiveness of this system to study automatically road user behaviour and safety.

2.2 Scene Interpretation

Scene interpretation applied to transportation seeks to understand traffic behaviour and analyze activities in the scene. One accepted framework for scene interpretation was introduced by Makris and Ellis (2005) where the scene is modeled as a topological map consisting of nodes and edges. The nodes are points of interest (POIs) which refer to regions where some activity occurs, whereas the edges are the activity paths which represent how the objects move between the POIs. Therefore, the activity analysis can be summarized in two main tasks; discovering the points of interest (POI) and learning activity paths. Activity learning is performed based on two main approaches: the

popular, trajectory learning and the recent, topic modelling (see (Hu, Tan, Wang, & Maybank, 2004; Morris & Trivedi, 2008a; Morris & Trivedi, 2013) for more details).

2.2.1 Point of Interest (POIs) Learning

The POIs correspond to the origins and destinations of tracked trajectories which refers mainly to the entry/exit zones and stop zones. There is relatively little work in the literature addressing the learning of POIs. Stauffer (2003) proposed a method to learn the entry and exit zones which was called sources and sinks in his work. The zones are learnt using a hidden Markov model (HMM) where all sequences are of length two, there are only two states (entry and exit) and the state model is not shared across time. A HMM represents each trajectory with hidden states and a transition matrix. Because of noisy tracking data, the author proposed a method for stitching broken/incomplete tracks. Given that this method assumes that all tracks are broken, a transition matrix is created using all pairwise transition likelihoods. Then, any two corresponding tracks should be stitched. In this work, detection of entry/exit zones and stitching tracking are conducted and updated simultaneously using an iterative process. Although this work described the applicability of stitching tracks, no information is given about the performance and effectiveness of the stitching. In addition, the author recommended that a thorough quantitative analysis is required for the stitching procedure. The dependency of zones detection and stitching could be an issue as well, i.e. the errors in zone detection may result in errors in the stitched trajectories and vice versa. McKenna and Nait-Charif (2004) modeled the POIs zones using Gaussian Mixture Model (GMM) and the Expectation Maximization (EM) algorithm. They used tracking data from a supportive home environment. Wang, Tieu, & Grimson (2006) learnt the entry/exit zones using trajectories extracted from far-field visual surveillance. Firstly, they classified the trajectories into two classes: pedestrians and vehicles based on the size. Secondly, the trajectories for each class are clustered into activity groups. Finally, the endpoints of trajectories of each activity group are modelled using GMM to learn the entry/exit zones. The fragmented tracks result in false entry/exit points at the endpoints of any incomplete track, hence, the authors remove these points from the dataset using a local density-velocity map before learning the zones.

In (Makris & Ellis, 2005), the authors used the trajectory endpoints to construct the entry and exit datasets which are modeled using GMM and EM algorithm. Two types of noise may be mixed with information of interest (actual POIs): noise caused by tracking failures and semi-stationary motion

noise generated by sources such as waving trees and water reflections: these are usually learnt as a separate GMM (noisy clusters) that can be distinguished using a density criterion. The same methodology was applied to a simulated intersection dataset in (Morris & Trivedi, 2011): the authors used the detected entry and exit zones to filter the trajectory dataset into two datasets of complete and incomplete trajectories. Only the complete trajectories were used in the motion learning phase, which may represent an important loss of some typical motion information (e.g. left turning users stopped in the intersection waiting for a gap in the opposite through traffic).

Recently, Nedrich and Davis (2013) modeled the entry and exit zones using "weak" trajectories, i.e. from trackers that provide multiple and frequently fragmented tracks per target. They applied a modified mean-shift clustering algorithm to cluster weak trajectory into higher level entities. The entity trajectories are then used to detect entry and exit regions based on the standard mean-shift clustering algorithm. Although the authors assert that mean-shift clustering is able to localize cluster modes automatically without knowing the number of clusters, the algorithm depends on a bandwidth criterion that is as challenging to select as the number of clusters. They used the detected zones to identify static occlusion regions based on an analysis of the distance distributions between the regions. From the available literature, it is evident that the GMM and the EM algorithms have been quite successful for learning interest points. It was tested mainly on cases of indoor scenes and limited work has been done in real traffic scenes. In addition, detected POIs can have a positive potential for further applications such as identifying noisy trajectories (e.g. incomplete trajectories) and cleaning datasets before the learning phase. Consequently, a consistent framework of the detection and applications of POI need further investigations.

2.2.2 Behaviour Analysis based On Trajectory Learning

The objective of trajectory learning is to cluster a dataset of observed trajectories into the main subsets of similar trajectories or motion patterns. Trajectory learning is a popular approach for scene interpretation and behaviour analysis. This approach has gained the researchers' interest because it fits into a typical surveillance and computer vision architecture as a higher level of the analysis where the trajectory of each object is detected and tracked at a lower level. This kind of analysis has three steps: pre-processing, clustering, and modelling. In some cases, the pre-processing step can be merged into the clustering step.

The preprocessing step is performed to set up trajectories suitable for standard clustering techniques which operate only on series or vectors that have equal length. Typically, the trajectories do not share the same length, even they move along similar paths, for many reasons such as the variation of individuals' behaviour, the differences in moving objects speeds and the tracking performance. To ensure equal lengths, the simplest and most popular technique is the normalization technique. Four approaches are used to normalize trajectories:

1) Resampling (e.g. linear interpolation);
2) Padding with repeat last position (assuming the velocity equals zero);
3) Extrapolating tracks with constant velocity model, these retain temporal information but predicts the future movement without any reasonable basis. Hence, the extrapolated tracks may be inaccurate;
4) Smoothing procedure to remove outliers; then interpolate and resample the trajectory.

The normalization technique is performed on all the trajectories whether they are complete or incomplete trajectories. Moreover, this technique discards the velocity information and distorts the data. Accordingly, it is recommended to avoid any pre-processing procedure in the trajectory leaning analysis and to search for a method can deal with the data as it is.

The clustering is the important step of trajectory learning. The key component of clustering is how to compute the similarity/distance between any pair of trajectories. The Euclidean distance is popular, but it requires that both time series have the same length and it is sensitive to distortions (e.g. shifting along the time axis). Therefore, the pre-processing step is mandatory before using the Euclidean distance. The development of elastic distance and similarity measures, such as Dynamic Time Warping (DTW) and Longest Common Sub-Sequence (LCSS), overcome the previous drawbacks. Both of DTW and LCSS are implemented using dynamic programming. DTW attempts to find the best alignment between two time series by minimizing the distance between them. Conversely, LCSS finds the length of the longest matching subsequence by comparing every point of the two time series using a given matching method. One limitation of the elastic measures is the high computation cost to construct a pairwise similarity matrix. Morris and Trivedi (2009) evaluated different similarity measures (the Hu distance computed as the average Euclidian distance between points on two trajectories (Hu, Xie, Fu, Zeng, & Maybank, 2007), PCA (Principle Component Analysis), DTW, LCSS, PF (Piciarelli and Foresti (Piciarelli & Foresti, 2006)),

Modified Hausdorff) and clustering methods for trajectories as a first step to understand road user behaviour. After tests on six different datasets, the authors concluded that LCSS was consistently the top performer. Nonetheless, the LCSS does not consider well the rate of change for time-series profiles such as speed profiles. In (Saunier & Mohamed, 2014)[5], a new similarity measure based on LCSS, known as aligned LCSS (ALCSS), is proposed to overcome the shortcoming of LCSS.

The second component of clustering is the selection of clustering algorithm. Clustering is the task of grouping similar elements from dataset in an unsupervised fashion. Among clustering algorithms, the hierarchical, partitioning, density based, and grid based algorithms are the best-known (Berkhin, 2006; Xu & Wunsch, 2005). A survey of time series clustering is presented in (Liao, 2005) for further reading. Partitioning algorithms are the most popular technique for their simplicity. (e.g. standard K-mean and its soft variant of Fuzzy C mean (FCM)). They work based on the following procedure: 1) selecting initial K (or C) centroids, 2) assigning each element to nearest centroid, 3) updating the centroids till the assignments no longer change (convergence achieved). To compute the centroids, these partitioning algorithms operate only on equal length time-series where the Euclidean distance may be used. Another clustering algorithm used for trajectories learning is spectral clustering. A spectral clustering algorithm is presented in (Ng, Jordan, & Weiss, 2002) that can be carried out easily using standard linear algebra libraries. This algorithm was tested on a number of challenging clustering situations, showing that spectral clustering performs better than traditional clustering algorithms. The advantage of spectral clustering methods is that it makes no assumptions on the distribution of data points. Instead it relies on the eigen decomposition of a similarity matrix which approximates an optimal graph partition. Therefore, the similarity/distance matrix can be constructed using any distance function (e.g. LCSS), then any traditional clustering (e.g K-mean) can be applied on top eigen vectors matrix derived from computed similarity matrix. Morris and Trivedi (2011) have successfully learnt the vehicle trajectories using spectral clustering algorithm based on LCSS and FCM. Spectral clustering is fast and takes as only input the predetermined number of groups. In our experience,

[5] In this reference, the ALCSS measure is proposed by Nicolas Saunier. This measure is used in this thesis for analyzing the speed profiles and safety indicators as a new application and one of the thesis contributions. The author of this thesis is a co-author in (Saunier & Mohamed, 2014).

finding the number of clusters by trial and error proved to be a challenge, and the resulting clusters are not always easy to interpret (Saunier & Mohamed, 2014).

Another popular clustering algorithm is the Hierarchical Clustering Analysis (HCA). HCA works by grouping similar trajectories into a hierarchical structure and presents the results in form of a tree-like structure (called dendrogram). There are two types of HCA: agglomerative and divisive based on bottom-up and top-down strategy respectively of constructing the dendrogram. The most used is the agglomerative HCA where each trajectory starts in its own cluster, and clusters are then merged until all trajectories are merged in one cluster or a defined cut line is reached. The dissimilarity of different clusters at each step is specified as a function of the pairwise distances of trajectories and on one of following linkage rules: single, complete, average and ward. Wen, Li, and Xiong (2011) extracted behaviour patterns using HCA. A new linkage rule, called length-weighted linkage, is proposed to distinguish between incomplete and complete trajectories. That linkage rule considers the effects of trajectories lengths. The HCA works based on a pairwise distance matrix which can be computed using any appropriate distance measure such as LCSS and DTW. The disadvantage of HCA is the inability of incremental learning (updating clusters when adding new data). HCA cannot adjust the dendrogram once it is constructed and it has to be rebuilt from the beginning. This is an issue in the traditional partitioning clustering as well.

The resulting clusters are modeled to find a representative for each cluster: this is performed during clustering (e.g. K-mean) or after clustering (e.g. spectral clustering). Many cluster algorithms such as K-means represent each cluster with the centroid of the cluster i.e. the average of trajectories in a cluster. To average trajectories of different lengths, they must be normalized before or after clustering to be of fixed length, which leads to lose the velocity information. An extension to the centroid model uses an envelope to represent each cluster based on its variance and extreme values along a path (Makris & Ellis, 2005) or a Gaussian distribution which is efficiently learnt using HMMs (Morris & Trivedi, 2011). HMMs are a possible clustering technique that is able to encode the spatio-temporal information probabilistically. Another cluster representation is a cluster prototype (e.g. the largest trajectory of each cluster (Saunier & Mohamed, 2014)). The choice of processing the trajectories in their original shape with variable lengths rules out for example classical clustering algorithms such as k-means since the concept of a centroid is not defined.

Some efforts have been made in behaviour understanding and its applications to anomaly detection and future behaviour prediction. Bennewitz et al (2005) studied motion patterns of people in a scene and updated the behaviour of a robot accordingly using the EM algorithm for clustering. Hu W., Xiao, Xie, and Tan (2004) modeled road user activities with a fuzzy self-organizing neural network. One of the main applications of this research is future motion prediction, which is highly influenced by the clustering. Another work by (Hu, et al., 2006) proposed a system for learning motion pattern hierarchically using spatial and temporal information with the fuzzy K-means algorithm, where the motion patterns are represented by a chain of Gaussian distributions. Anomaly trajectory and behaviour prediction were performed based on the motion patterns. Atev, Masoud, and Papanikolopoulos (2006) modeled traffic patterns in an intersection using a modified Hausdorff distance and spectral clustering. Recent work by Morris and Trivedi (2011) focused on automated activity learning in an unsupervised fashion. The authors proposed a three-stage hierarchal learning framework to analyze object activities and predict future activities, as well as to detect abnormal events. The authors used the LCSS as the trajectory similarity measure and spectral clustering for trajectory clustering.

2.2.3 Behaviour Analysis based On Topic Modelling

Recently, significant research has been done for behaviour analysis based on topic modelling. Topic modelling was first developed in natural language processing with algorithms such as probabilistic Latent Semantic Analysis (pLSA) (Hofmann, 1999) and Latent Dirichlet Allocation (LDA). (Blei, Ng, & Jordan, 2003). The idea of topic modelling is to view any document as a mixture of various topics where distribution of words generates the different topics. It aims to cluster the words that co-occur in the same document into the same topic. Both of pLSA and LDA require to define the number of topics in advance. Furthermore, Hierarchical Dirichlet Process (HDP), an extension of LDA, is a nonparametric Bayesian model proposed by (Teh, Jordan, Beal, & Blei, 2006) to automatically learn the number of topics in any training dataset using Dirichlet Processes (Ferguson, 1973) as priors. The readers are referred to (Wang, 2011) for more details about topic models for action recognition.

To adopt topic models for activity learning, three items in language processing (documents, words, and mixture of topics) should be modeled. Given an input video, there are two forms of input data

for topic modelling: optical flow vectors (Fu, Wang, Li, Lu, & Ma, 2012) and trajectories (the output of video tracking) (Wang X. , Ma, Ng, & Grimson, 2011).

Trajectories could be treated as documents and observations on the trajectory as words (Wang, Ma, & Grimson, 2009). A word is defined as three dimensional vector (x, y, dir); where x,y is the position and dir is the moving direction (derived using the velocities). A codebook is created by dividing any scene into small regular cells (one or more pixels) and is used to quantize each word. Hereafter, topic model is applied to cluster co-occurring words into one topic. Each topic is mainly characterised by its moving direction. This topic represents the subset of any path, known as a semantic region and a combination of semantic regions represents an activity. Accordingly, Wang, Ma, and Grimson (2009) developed a Dual-HDP model which applied the HDP in two layers. The first layer clusters the motion into semantic regions and the second layer clusters semantic regions into activities. Using Dirichlet Processes (DP), the whole learning procedure is unsupervised where Dual-HDP model can define the numbers of both semantic regions and activities directly from training dataset. Wang et al (2011) extended the use of Dual-HDP to Dynamic Dual-HDP for allowing dynamic updating of the activities model (incremental learning) and on-line detection of abnormal activity.

Saleemi, Shafique, & Shah (2009) learnt the motion patterns in form of a multivariate nonparametric probability density function (pdf) of spatio-temporal variables. The pdf is learnt using kernel density estimation. Fu et al. (2012) learnt motion patterns based on an improved sparse topical coding framework. The video is segmented into a sequence of clips (documents). To generate the words, optical flow vectors are extracted for each consecutive frames and quantized into a codebook as position information and direction information (8 directions are assumed in this work). Song, Jiang, Shi, Molina, and Katsaggelos (2014) proposed a two-level motion pattern learning using the basic LDA formation. A video recording is divided into video clips as documents and visual words represent the patch-based motion features, including position, direction, and speed. The first layer models the activities for a single-road user, while the second level uses the output of first layer as words to learn the interaction patterns by analyzing the typical occurrences of multiple objects at the same time.

Topic models rely on local motion, so they are robust for fragmented trajectories due to occlusions and tracking error. Moreover, they can be used to detect anomalies not only in single road user

behaviour but also based on interaction behaviour. On the other hand, they ignore the temporal order of observations of a trajectory which makes them unsuitable to accurately assess future behaviour for each individual road user, and consequently unsuitable for surrogate safety analysis.

Instead, trajectory learning provides an appropriate description of typical motions. However, existing trajectory learning methods have several shortcomings as follows:

1. The number of clusters / activities is required in advance in most of clustering methods.
2. Tracking dependency: the quality of trajectory learning is influenced by the tracking output. Good tracking results in robust learning while noise or errors in tracking leads to bad quality motion patterns.
3. The construction of the pairwise similarity matrix is computationally expensive in both of time and required storage space for large dataset with non-metric similarity measures (e.g. LCSS). Computing similarity between two trajectories using LCSS requires to construct an internal matrix for the matching between all points in each trajectory (or up to a defined bound (Vlachos, Kollios, & Gunopulos, 2002)). The bound is defined to control how far in time a point in a trajectory can match a point in the other trajectory. This internal matrix is computed $\frac{(N^2-N)}{2}$ time for a training dataset with N trajectories to construct the full pairwise similarity matrix.

2.2.4 **Speed Profiles Analysis**

The speed profile is derived from a sequence of positions (spatial information) through differentiation and computing the norm of the velocity vector. Most previous work on behaviour understanding focuses only on spatial information considering that speed profiles are already embedded in the analysis. Speed profiles vary along the same path and analysis of individual speed profiles is hence useful for behaviour understanding. Limited work exists to cluster individual speed profiles directly, separately from motion patterns. Parkhurst (2006) examined the shape of speed profiles to understand the driver behaviour at urban and rural non-signalized intersections. A typical speed profile was studied in the case of a vehicle coming to a complete stop or failing to complete their stop at a stop sign intersection. Laureshyn, Åström, & Brundell-Freij (2009) classified the speed profiles, extracted from automated video data, of vehicles making left turn at a signalized intersection. The left turn manoeuvre mainly interacted with two different types of

road users: oncoming traffic and pedestrians at the pedestrian crossing. The authors used three types of pattern recognition techniques: k-means (unsupervised), k-nearest neighbors (supervised), and dimension reduction (singular value decomposition). They concluded that the three pattern recognition techniques perform well with accuracy equal to 81 % to 91 % compared to human manual classification, with the advantage of being automated. Nevertheless, the joint or hierarchical learning of motion patterns and speed profiles and their use for behavior prediction have not been yet investigated.

2.3 Motion Prediction and Collision Avoidance

The choice of a method for motion prediction is crucial to evaluate whether road users are on a collision course and to compute several safety indicators. Such methods are very similar to navigation and path planning in robotics, where collisions should be predicted and avoided. The difference is that robots know their goals, in particular places to reach, and can plan accordingly, while the analysis of road user interactions based on exterior observations does generally not have access to their internal state and goals. Nevertheless, path planning requires taking into account all obstacles and the movement of all other moving objects, i.e. it relies on motion prediction methods for the assessment of the safety of planned movements, just as surrogate safety analysis. The goal of this subsection is therefore to give an overview of the state of the art of methods for motion prediction in the field of robotics and surrogate safety analysis.

The early work of Zhu (1990) describes three types of motion prediction models:

- the constant velocity model: it assumes that the object moves with no change in speed nor direction;
- the random motion model: it assumes that acceleration changes according to probability distribution. That means that the motion is controlled by drawing the acceleration randomly and independently at each step from the defined distribution;
- the intentional motion model: the objects move in a scheduled way (e.g. an object moves towards a specific goal and/or seek to avoid collision).

These three models fall into two categories, the deterministic and stochastic motion prediction approaches (Eidehall & Petersson, 2008):

Deterministic motion prediction consists in predicting a single future trajectory for each object. The constant velocity model is one such method, choosing the most probable trajectory among several alternatives is another. The former approach has been the default, sometimes implicit, method used to compute several safety indicators such as the TTC (Hydén C. , 1996; RSM, 2003; Cunto, 2008; Ismail, 2010; Laureshyn A. , 2010).

Stochastic motion prediction relies on taking many different scenarios into account. With the rise of computer power, this approach is becoming more manageable and therefore more popular. In robotics (Thrun, Burgard, & Fox, 2005), while the state and goals of the robot are known, the movement of other objects is usually modeled stochastically. There are several stochastic motion prediction methods, among which one can cite: vehicle motion model using Monte Carlo simulation (Broadhurst, Baker, & Kanade, 2005; Eidehall & Petersson, 2006; Danielsson, Petersson, & Eidehall, 2007; Eidehall & Petersson, 2008), reachable sets (Althoff, Stursberg, & Buss, 2008), Gaussian processes (Laugier, et al., 2011) and trajectory learning (Hu, Xiao, Xie, & Tan, 2004; Bennewitz, Burgard, Cielniak, & Thrun, 2005; Saunier, Sayed, & Ismail, 2010; Morris & Trivedi, 2011)

Motion prediction based on driver preference distributions has been examined previously in the literature (Broadhurst, Baker, & Kanade, 2004; Broadhurst, Baker, & Kanade, 2005; Eidehall & Petersson, 2006; Eidehall & Petersson, 2008; Danielsson, Petersson, & Eidehall, 2007; Sorstedt, Svensson, Sandblom, & Hammarstrand, 2011). In a traffic scene, a general threat assessment using Monte Carlo simulation was first proposed in (Broadhurst, Baker, & Kanade, 2005) and extended in (Eidehall & Petersson, 2008). Broadhurst et al (2005) proposed a reasoning framework for the future motion of multiple objects in a scene. Vehicle motion is predicted by using the current positions and velocities of all objects as known variables and by modeling the future behaviour control inputs (e.g. acceleration and steering) as random variables. The random control inputs, which determine the future trajectories, are generated in a Monte Carlo Simulation. As a result, the probability of collision can be assessed for a path given the initial state and a particular control input. This is useful to determine which future control inputs are safe for a robot to achieve its predefined goal safely. Eidehall and Petersson (2008) refined the previous trajectory generation framework using an iterative sampling process which aimed at removing and replacing samples that cause collisions at an early stage of simulation. In addition, visibility constraints were added to capture the fact that the driver is more attentive to the front of the vehicle than to other directions.

Sorstedt et al (2011) proposed a new vehicle motion model taking into account four components of a cost function to select a trajectory: longitudinal velocity, lateral positioning, comfort of trajectory, and vehicle interaction.

Althoff et al (2008) predicted the possible future behaviour of traffic participants using a probabilistic framework and stochastic reachable sets. The longitudinal dynamics of the vehicles are modeled by a hybrid automaton combining discrete and continuous dynamics. Four modes describe the discrete dynamics: acceleration, braking, standstill and speed limit. The reachable sets are computed using Markov chains. Similarly, Sekiyama, Minoura, and Watanabe (2011) decompose future behaviours into spatial regions called Attainable Regions (AR). A collision is predicted if two ARs overlap each other.

Laugier et al (2011) used both hidden Markov models (HMM) and Gaussian processes (GP) to estimate the probability of collision for a given vehicle. The authors advocate the use of TTC as a common indicator to assess the collision risk. The architecture of their model is composed of three sub-modules: 1) a driving behaviour recognition module using HMM to estimate the probability distribution of driver behaviour, 2) a driving behaviour prediction module using GP, and 3) a risk estimation module. Lambert , Gruyer, Pierre, and Ndjeng (2008) represented the movement of each robot and obstacle as a Gaussian distribution. The probability of collision is computed from the integral of the product of the distributions, taking into account the configuration and volume of the objects.

The last approach relies on learning trajectories for prediction. The approach relies on the fact that motion is usually not random but structured and repetitive. The method comprises two phases: the learning and the estimation phases. As stated in the scene interpretation section, the learning phase clusters a training dataset of observed trajectories into the main motion patterns. The estimation phase estimates the probability that a new observed trajectory will follow a learnt motion pattern. This approach has the advantage of representing the environment well and allowing long-term prediction. Bennewitz et al (2005) learnt the motion patterns of people in a scene which enables a robot to update its behaviour accordingly. The trajectories are clustered based on the Expectation Maximization (EM) algorithm. HMMs are used to estimate the persons' current and future movements. It was reported that the approach provided a good estimation of the persons' future movements and that it consequently improved the navigation process of mobile robots. Hu et al

(2004) learnt vehicle activities with a fuzzy self-organizing neural network. This method predicts the future activity (motion) of a vehicle according to its past motion. In this way, the probability of collision between two road users can be approximated by a discrete sum when taking into account a finite number of the most probable predicted trajectories.

It can be concluded from this literature review that most approaches for motion prediction are probabilistic in order to take into account the intrinsic uncertainty of the task. There is a wide variety of methods and it should also be noted that there is a relationship between the prediction method and the representation of motion, e.g. a discrete trajectory, a mathematical function or probability distributions, which in turn conditions what can be done with the predicted motion and what indicators can be computed.

2.4 Safety Analysis

2.4.1 Collision-based Safety Analysis

Traditionally, collisions are the main indicators of road safety analysis where a collision is defined as a sequence of events lead to different scales of injuries or property damage of one or more road users. Collision data is collected from the reports provided by police, insurance company, and emergency vehicles interventions. The safety analysis is conducted based on collision frequency and/or collision severities in order to identify “unsafe” sites for treatment and to find the right counter-measures. Collision frequency is defined as the number of collisions occurring in a specific site per unit of time while collision severity refers to injury level or property damage only.

Road safety diagnosis aims to understand the factors associated with the “cause” of collisions and their severity. These factors could be related to characteristics of the vehicle, the infrastructure or the road users and they can have an influence on the series of events before, during, and after the collision. Pre-collision events increase the probability or the severity of a collision (e.g. inattention, worn brakes, wet pavement, etc). Events during the collision influence the severity of collision (e.g. driver speed, seat belts, airbag, etc.). Post-collision events depend on the efficiency of emergency services (e.g. time of emergence response). Although collision based safety analysis is the most common method for estimating safety, it has several limitations that have been extensively discussed in the literature, e.g. in (Elvik, Vaa, Erke, & Sorensen, 2009; Ismail, 2010; Laureshyn A. , 2010). Collision data limitations can be summarized as follows:

- *Not all collisions are reportable:* The main reason is the definition of minimum threshold to report a collision which changes from country to country. For examples, some countries report all property damage only (PDO) collisions, while others report PDO collisions only if exceeding a certain cost threshold (e.g 1000$). This threshold should be adjusted from time to time to cover inflation. Moreover, if a collision involves an injury, a fatal collision is defined if one or more persons injured in the collision dies within the following 30 days. Therefore, the improvement of medical services could make the person live more than the 30 days, although it is not an indication of an improvement of road safety.
- *Not all reportable collisions are reported:* some cases reportable to the police are not reported for many reason such as the absence of legal obligation to report the collision, the lack of awareness of injury at the collision time, and the avoidance of insurance company penalties (or other type of penalties). Single–vehicle collisions have lower reporting rates. Moreover, vulnerable road users (e.g. pedestrian and cyclist) are underrepresented in the police report compared with other data such as hospital data (Svensson Å. , 1998). To estimate the proportion of unreported cases, the comparison between different collision sources such as police and hospital reports can be conducted.
- *Collisions are rare events:* They are therefore associated with the random variations inherent in small numbers. It is not sufficient to use collision data only for a short period of time: 3 years of collision data are typically necessary. Another issue caused by rarity is that many potential factors change over these long periods of analysis: it is the problem of attribution of the change in safety.
- *Safety analysis based on collision records is a reactive approach:* Road safety improvement programs depend on the occurrence of a sufficient number of traffic collisions to analyze and take action.
- *Understanding collision processes is limited:* police officers mainly investigate the responsibility of a collision rather than the causes and the processes that lead to the collision. As a result, collision reports provide little information about the process of a collision: they are intrinsically ill-suited to understand the mechanism or chain of events leading to the collision. Collision reconstruction is one of the methods to understand collision processes. It requires a safety expert to collect the volatile information before

removing the vehicles involved in the collision. Therefore, collision reconstruction is difficult and very expensive to perform (Laureshyn A. , 2010).

2.4.2 Methods for Surrogate Safety Analyses

Methods for surrogate safety analysis consist in assessing the safety of a given site without waiting for collisions to happen. Surrogate measures rely on the observations of the road users interactions without collision. The surrogate measures should satisfy two conditions to be used in place of observed collision (Tarko, Davis, Saunier, Sayed, & Washington, 2009):

i. Surrogate measures of safety should be able to observe interactions without collision which have the same mechanisms as collisions.

ii. Surrogate measures should have a causal link to predict the expected frequency and/or severity of collisions.

Furthermore, surrogates measures should occur more frequently than collisions (Archer, 2005). There is a large and growing body of literature on methods for surrogate safety analysis. Readers are referred to these PhD theses (Svensson Å. , 1998; Archer, 2005; Laureshyn A. , 2010) and the Transportation Research Board (TRB) white paper (Tarko, Davis, Saunier, Sayed, & Washington, 2009) for more details.

2.4.2.1 Surrogate Safety Indicators

For surrogate safety analysis to be objective, a number of quantitative safety indicators have been proposed in the literature to measure the proximity to a potential collision, or probability of collision, and the severity of the potential collision. A recent review of safety indicators yielded a set of 119 unique indicators, where 98 indicators (i.e. 86 %) are calculable using mainly the position data (Laureshyn A. , 2010). Common safety indicators will be discussed in the following sections.

Time to Collision

Time to Collision (TTC) is one of the most popular time-based safety indicator which is defined for a given motion prediction method as the time required for two vehicles to collide following the predicted trajectories. If several predicted trajectories are available, with corresponding probabilities, the expected TTC can be computed (Saunier, Sayed, & Ismail, 2010). Hence, TTC values are between 0 (in case of collision) and infinite (when no collision course exists). TTC

measures the proximity of a potential collision where the lower the TTC, the nearer the road users to a potential collision (Hayward, 1972).

Time to Accident and Conflicting Speed

Time to Accident (TA) was proposed for the Swedish TCT by (Hydén, 1987) as the TTC just before one of the road users attempts an evasive action such as braking. In the Swedish TCT, TA and Conflicting Speed (CS) are used to estimate the conflict severity level. The TA/CS diagram is shown in Figure 2.1. Conflicting Speed (CS) is the speed of the road user who took an evasive action at the instant of measuring TA.

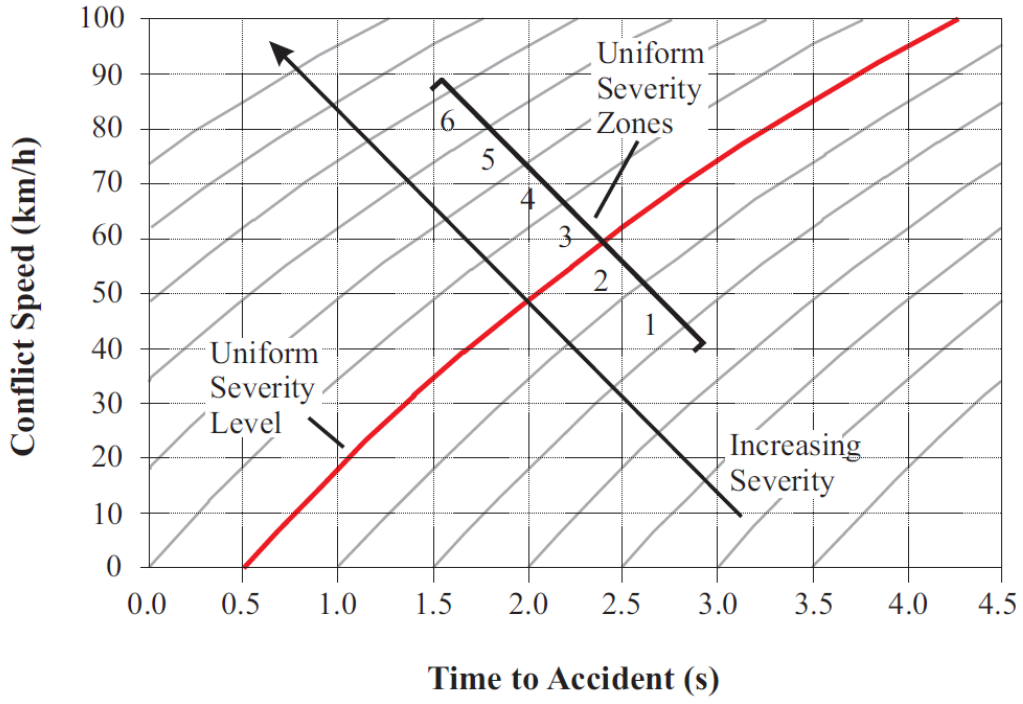


Figure 2.1: TA/Conflicting speed diagram to measure interaction severity level (Archer, 2005)

Post Encroachment Time

Post Encroachment Time (PET) is another time-based safety indicator, defined only in cases where the road users' observed trajectories cross as the time difference between the instants at which the two vehicles pass the crossing zone, see Figure 2.2, (Allen, Shin, & Cooper, 1978). PET is very different in nature from TTC since it is based on the complete observed trajectories and only one value may be computed. The concept of PET was extended to be computed continuously at each instant as the PET for the trajectories predicted for the road users, given a motion prediction

method. This indicator was first proposed in (Hansson, 1975) (cited in (Laureshyn A. , 2010)) and is sometimes called gap time (GT).

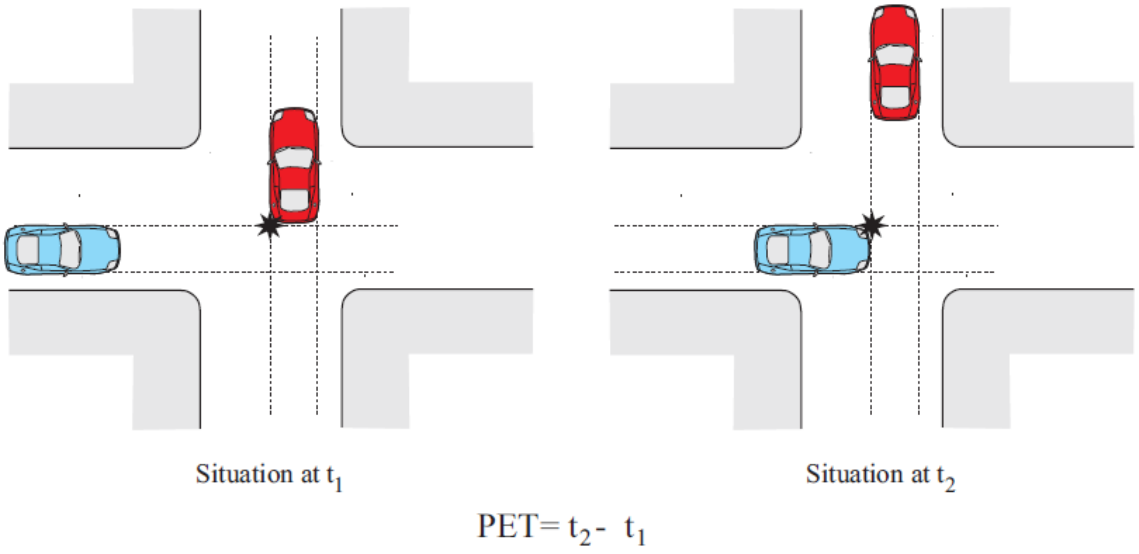


Figure 2.2: The Concept of post-encroachment time (PET) (Cunto, 2008)

Interpreting Interactions and Safety Indicators

In most TCTs, a specific value of a continuous safety indicator (e.g. TTC_{min} and TA) is used and compared to a threshold to distinguish, usually for diagnosis purpose, the most severe conflicts from "safer" interactions. The Federal Highway Administration (FHWA) designed the piece of software Surrogate Safety Assessment Model (SSAM) to perform the analysis of trajectory data extracted from microscopic simulation software (Gettman, Pu, Sayed, & Shelby, 2008). SSAM uses a predefined threshold for different safety indicators to identify the most severe conflicts among all road user interactions (e.g. the default threshold on minimum TTC is 1.5 s). The most severe value of safety indicators is typically used to summarize them, for example minimum values for spatio-temporal indicators (e.g. distance, TTC and GT) or maximum values for probability of collision or deceleration to safety time. However, as argued in (Laureshyn, Svensson, & Hydén, 2010), narrowing down the whole interaction to a single value leads to losing a lot of information.

A development of the TTC measure has been suggested by (Minderhoud & Bovy, 2001) to overcome the problem of using a single value of TTC. The authors suggested two safety indicators based on TTC. The indicators are calculated using vehicle trajectories collected over a specific duration (H) for a specific road segment (L). The first proposed indicator is the Time Exposed TTC

(TET indicator) which measures the duration of exposure that all vehicles involved in conflicts spend under a designated TTC minimum threshold during a specified time period. The second indicator is the Time Integrated TTC (TIT indicator) which uses the integral of the TTC profile of drivers to express the level of safety over the specified time period. Figure 2.3 illustrates the concept of TIT and TET. The problem with these indicators is that the selection of proper thresholds is arbitrary and these indicators still condense the whole indicator profile into a single measure through integration.

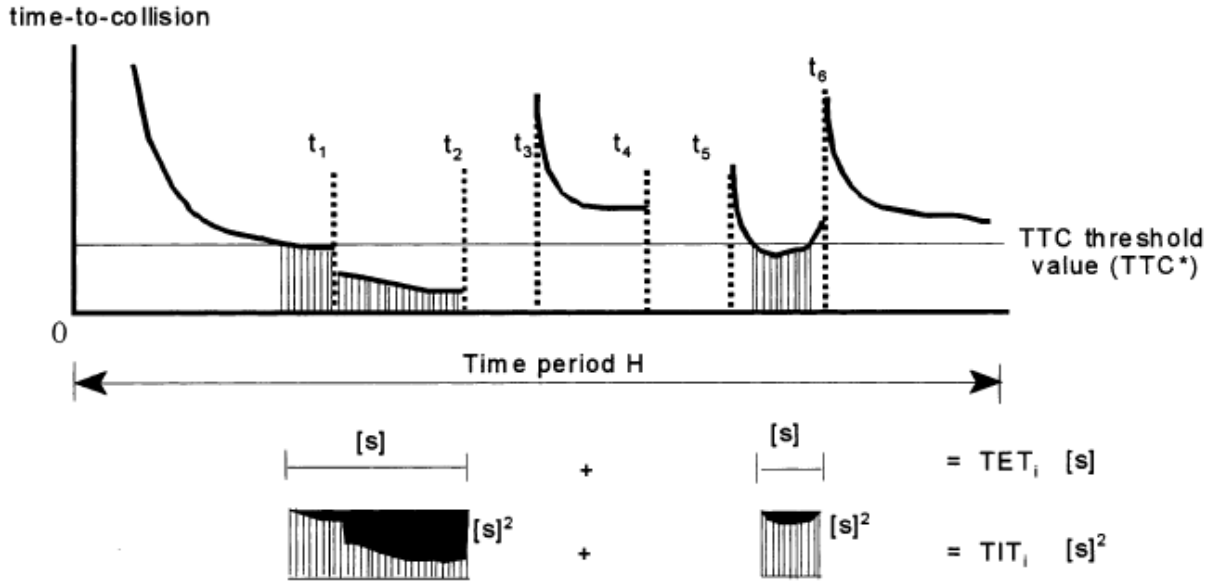


Figure 2.3: Concepts of Time-Exposed TTC and Time-Integrate TTC (Minderhoud & Bovy, 2001)

2.4.2.2 Surrogate Measures and Collisions Relationship

Numerous attempts have been conducted to find a stable relationship between surrogate measures of safety and collisions. This relationship could be used as a predictor of collision frequency based on the collection of surrogate measures of safety. A simple relationship is represented by the following equation (Hauer, 1982) :

$$A = B * C$$

Where A is the expected number of collision, B is the collision-to-surrogate ratio, and C is the number of interactions (e.g. conflicts). B is therefore the probability of a collision given an interaction. A crucial goal for the use of surrogate safety analysis is to validate the relationship

between interactions with and without a collision. The following section details the most common approaches to figure out the relationships between interactions and collisions.

Safety Continuum (Hierarchy)

In TCTs, the safety continuum plays an important role to rank all traffic interactions with regard to safety. It represents the relationship between the frequency of interactions and their severity. This model can provide a complete picture of interactions which may help to understand collision mechanisms. Archer (2005) define the safety continuum as:

> *"Theoretical concept inferred in relation to the use of proximal safety indicators whereby all interactions are placed on the same scale with safe passages at one extreme and accidents involving fatalities at the other"*

Hydén (1987) defined the safety hierarchy as a pyramid-like model with accidents at the top, the normal "undisturbed" passages at the bottom, and traffic conflicts between them. Accidents are classified into fatal, injury, or property damage only (as shown in the right of Figure 2.4). Conflicts are defined based on the existence of collision course at constant velocity and evasive action. Conflicts thus do not include the normal traffic behaviour such as interactions between two road users where there is no collision course at constant velocity and predicted trajectories cross each other with no potential collision point. Svensson (1998) and Svensson & Hydén (2006) argued that the "true" shape of the safety continuum is a diamond-like shape and not a pyramid (as shown in the left of Figure 2.4). When the conflict distribution is analyzed, the author found that less severe interactions are quite rare while the majority of interactions are of medium severity.

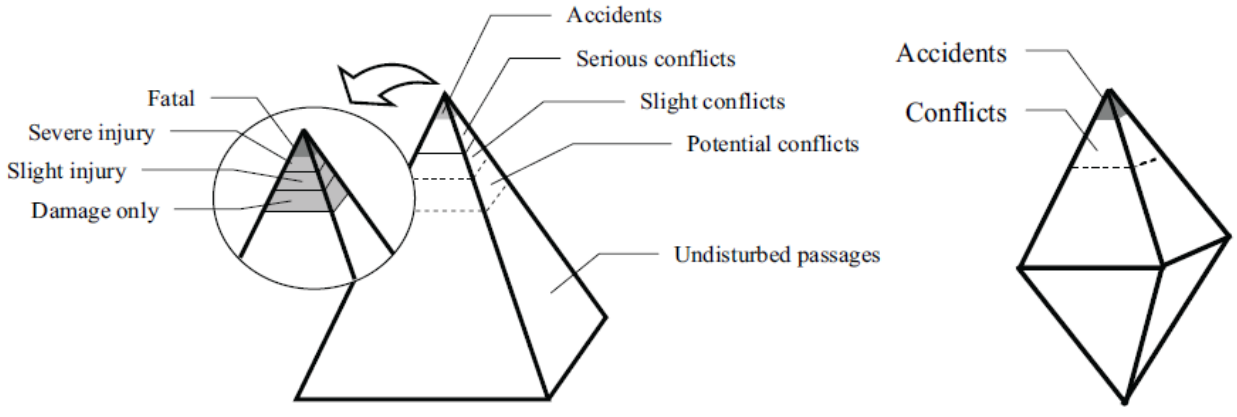


Figure 2.4: Safety Hierarchies Examples: pyramid model (on left) and diamond model (on right) (Laureshyn A. , 2010)

Recently, Zheng, Ismail and Meng (2014) mapped the safety continuum as a shifted Gamma-generalized Pareto Distribution (GPD) model to map the whole safety continuum. This model is used to estimate the number of collisions related to lane changing maneuvers on freeways. This model has the advantage of eliminating of the dependency of thresholds selections and considering the relationship between conflicts and collisions as non-linear.

Extreme Value Method (EVM)

Songchitruksa and Tarko (2004; 2006) introduced an approach to estimate the frequency of collisions based on the Extreme Value Theory (EVT). EVT is a statistical method which involves estimating the extreme values over short term available data. It enables the extrapolation of the unobserved value based on the distribution of observed values. This method solved the issue of using fixed surrogate-collision ratio by replacing it with a probability distribution. To apply EVT on safety analysis, Songchitruksa and Tarko (2006) measured the proximity to collisions by observing PET for right-angle interactions at signalized intersections. The PET value of zero represents the occurrence of collisions which is missing (unobserved values) in the PET measurement due to the observation of interactions without collision. By analyzing the distribution of observed PETs, EVT can estimate the expected potential number of collisions (PET less than zero) in the entire intersection. The authors evaluated the optimal observation periods based on simulation study and compared the results with actual crash data for 4 years. They concluded the need to collect a larger amount of field data (about 30 to 50 days) to construct a reliable distribution.

Counterfactual Approach

Another approach is called the counterfactual approach which understands the relationship between surrogate measures of safety and collisions. Davis, Hourdos, Xiong, and Chatterjee (2011) proposed an outline of a simple theoretical method based on causal model which represents collisions and interactions resulting from background (initial) conditions and evasive actions. The authors applied Pearl's theory (Pearl, 2000) of probabilistic causal model on a small set of collisions and surrogate measures. The causal model developed by (Pearl, 2000) consists of exogenous variables (a set of background conditions), and endogenous variables (evasive action) and the crash occurence. The probability of a collision depends on backgound conditions and evasive action (Davis, Hourdos, Xiong, & Chatterjee, 2011). The conditional independent structural model could be presented in the following form:

$$P(y,u,x) = P\,(y|x,u)P(x|u)P(u)$$

Where y is the collision outcome, x is background condition and u is the undertaken evasive evasive action or more generally user control/input.

To illustrate the concept of this model, the authors take rear-end collisions as an example due to its simplicity. The initial condition is defined for lead car (v_1= speed, a_1= acceleration) and for the following car (v_2=speed, r_2=reaction time, h_2=headway). The evasive action considered is the following car acceleration (a_2). It must be noted that different types of collisions will be described by different variables. The equation used in the rear end collision case is as follows:

$$y\,(crash\ output) = 1\ if\ 0 \geq v_2h_2 + \frac{v_1^2}{2a_1} - v_2r_2 - \frac{v_2^2}{2a_2}, 0\ otherwise$$

In this case; the critical value of y is zero. Through knowing the initial condition, the minimum evasive action (a_2) could be calculated as the minimum required braking value. The challenge in other collision types in particular with crossing paths is to specify the initial conditions required for building the structural model. In addition, the evasive action value must be calculated by simulation according to the initial condition values and it is challenging to know the distributions of all initial conditions.

In this vein, Davis et al (2011) found that the evasive actions undertaken by the road users involved in interactions without a collision differ from the ones attempted in collisions. They suggested the theoretical treatment of a causal model of interactions without collision and with collision. Three steps were suggested for future work:

i. Develop a structural model of crash events
ii. Understand how evasive actions are achieved
iii. Know the relative frequencies of the different conflict severities.

2.4.2.3 A Probabilistic Framework for Automated Road Safety Analysis

Another promising method for surrogate safety analysis is proposed by (Saunier & Sayed, 2008). The authors propose a probabilistic framework to extend traditional surrogate measures of safety such as TTC and to compute the probability of collision. The core concept of this method is to consider different possible future paths that each road user may follow: these paths and associated

probabilities are established by learning typical trajectories using machine learning techniques. Hu et al. (2004) described a system to use a probabilistic model to predict potential collisions. This system relies on learning vehicle activities using fuzzy self-organizing neural network methods. Likewise, Saunier and Sayed (2008) proposed a probabilistic framework using an automated vision-based system. The formula to compute the collision probability at an instant for two interacting road users was taken from the work of (Hu, Xiao, Xie, & Tan, 2004). The predicted positions are computed for a specified time horizon according to a defined extrapolation hypothesis for each road user by matching past trajectories with typical motion patterns and following the matched motion pattern.

A refined and expanded probabilistic framework was proposed by (Saunier, Sayed, & Ismail, 2010). A large dataset containing more than 300 severe conflicts and collisions is studied Normal situations are needed to investigate a complete safety hierarchy. This method needs further investigation.

2.5 Summary

The literature review examined some of the important research in surrogate safety analysis, scene interpretation, and motion prediction methods. It is noticed that there is an interdependency between the different topics. For instance, conducting surrogate safety analysis requires a robust motion prediction to compute several indicators. Motion predictions methods fall in two categories deterministic models (e.g. the popular method is constant velocity) and stochastic models (e.g. methods based on the learning of typical motion patterns). Scene interpretation topic is used to learn MPs which relies on the quality of extracted trajectories.

The literature review demonstrated the following shortcomings in past research, which will be addressed in this dissertation:

1) Video Analysis Technology

    a. Extracted trajectories are noisy: removing this noise can lead to a better analysis of behaviour.

    b. Including noisy trajectories (e.g. fragmented trajectories) is one of the limitations of trajectory learning which results in a more difficult and time consuming learning process.

2) Scene Interpretation

    a. Learning POIs and their transportation applications needs further investigation. POIs could help improve scene interpretation and behaviour analysis.

    b. Existing trajectory learning methodologies have several shortcomings such as the expensive computation of the pairwise similarity matrix, the dependency on tracking performance and the a priori information of the cluster number.

    c. Few works use the temporal information, in the form of individual speed profiles, to learn motion patterns.

3) Motion prediction methods

    a. Most surrogate methods for safety analysis are based on motion prediction at constant velocity, and further investigation of different prediction methods is required to compute surrogate safety measures.

    b. No comparison of the effectiveness of different motion prediction methods for computing safety indicators and performing the safety evaluation of a site.

4) Surrogate safety analysis

    a. Existing research focuses on finding relationships between surrogate measures of safety and collisions which have not yet been consistently established.

    b. A generic and consistent framework for reliable automated safety analysis is required by incorporating the development of the three topics: video analysis, scene interpretation, and motion prediction.

    c. The interpretation of continuous indicators relies on the aggregation of the whole time series into a single value which leads to losing a lot of information and is prone to errors coming from the outliers.

CHAPTER 3 RESEARCH METHODOLOGY (PHASE ONE) AN APPLICATION OF COMPUTER VISION TECHNIQUE FOR AUTOMATED SCENE INTERPRETATION

3.1 Overview

The main source of data acquisition for this research is video sensors which provide detailed microscopic information for all road users. Therefore, an automated or semi-automated video analysis tool must be used in the beginning to extracted trajectories from the video frames in large enough quantities. This chapter explains the first phase of our research which concerns the application of computer vision technique for automated scene interpretation and behaviour understanding. The following sections presents how TI tool works, how it is improved to fit the requirements of the scene interpretation (multi-level MP learning) framework to understand scene activities. An overview of the methodology is presented in Figure 3.1.

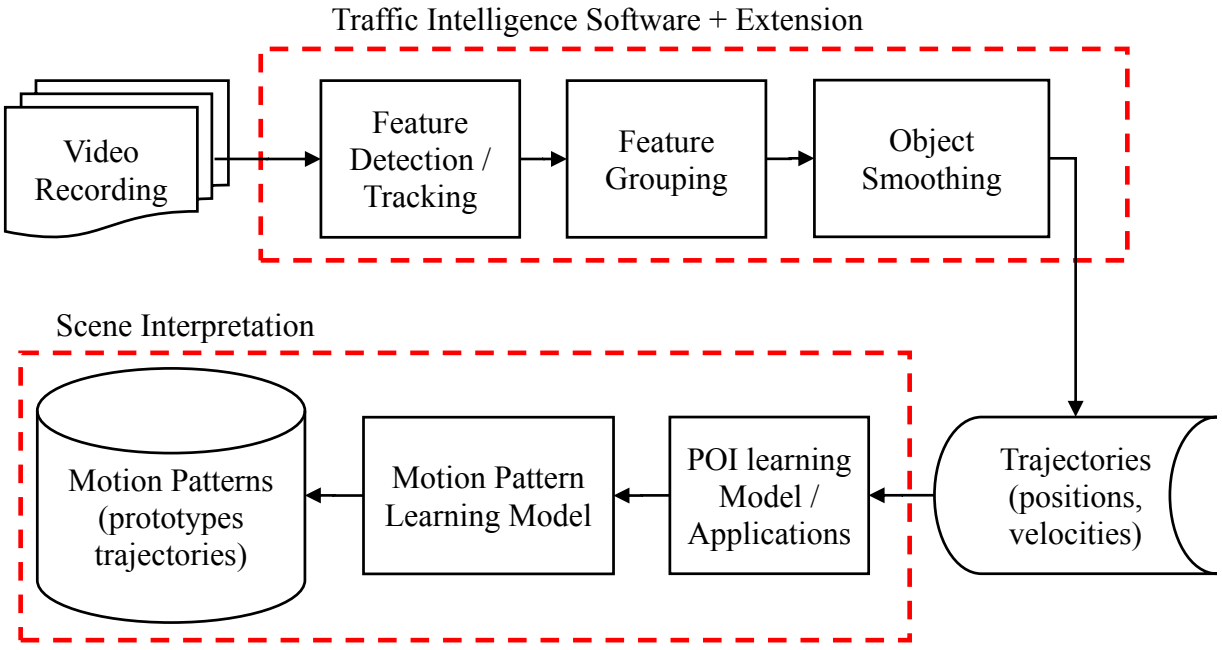


Figure 3.1: Phase One Methodology Overview

3.2 Computer Vision based for Obtaining Road User Trajectories

As a first step, a video tracking tool from the open source "Traffic Intelligence" (TI) project (Saunier N. , 2014) is used to detect and track moving road users: the result, road user trajectories,

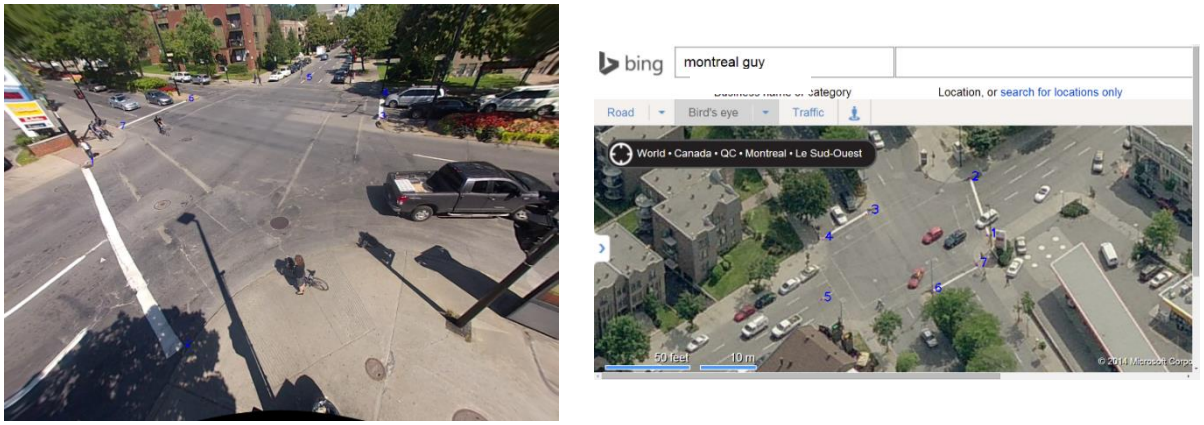
is the main input of the proposed methodology. The advantages of TI are that it is an accessible and automated road user tracking tool. TI analyzes video data based on a feature tracking algorithm which it is performed in two steps. First, all distinct points (features) that move (more than a defined distance) are tracked from frame to frame. The second step, the feature grouping algorithm, is to group feature trajectories with similar motion into an "object" hypothesis. In order to obtain trajectories in the "real world" and not in image space, a pre-processing step called camera calibration is required before tracking. The main steps of video tracking tool could be summarized into camera calibration, feature tracking algorithm, and feature grouping algorithm. Following sections describe each step.

3.2.1 Camera Calibration

The recorded video captures the real 3D world space projected into 2D image space. The proposed analysis whether for scene interpretation or surrogate safety analysis relies on accurate positional data of tracked road users in world coordinate system. Hence, recovering the positions of various features from image space to world space is required. This process, called camera calibration, is conducted by the definition of camera parameters. Camera parameters are composed of intrinsic and extrinsic parameters. The intrinsic parameters are the focal length, skew, and radial distortion of the lens which are necessary to correct the observation in the image space level. The extrinsic parameters describe the location and orientation of the camera.

In the TI tool, the camera calibration is performed by estimating a homography matrix. The homography matrix is a 3x3 matrix that represents the perspective transformation between image and world planes. Homography can be estimated by identifying at least 4 non collinear points such as distinct landmarks in the camera image and in the real world. The corresponding points can be obtained accurately using traditional surveying equipment (e.g. Total stations, GPS, etc) or simply obtaining an aerial view covering the site with a known scale (e.g. from Google Earth, satellite images, photogrammetry images). Hereafter, the user annotates distinct landmarks in the camera image such as pavement marking, curbs, and poles and corresponding points in the satellite image. When multiple points are obtained, an optimization algorithm based on nonlinear least square adjustment method is used to calibrate and compute the homography matrix. The procedure follows

the function "findHomography" in OpenCV[6]. In each camera view, different homography matrix must be defined. An example of selected image landmarks and corresponding world points is shown in Figure 3.2. Once the homography matrix is computed, trajectories are transformed to the real world coordinates.



a) Image space b) World space

Figure 3.2: An example of distinct landmarks in image space and corresponding points in world space

Some of the video data are captured using a GoPro HD Hero2 camera. This camera is inexpensive, small, and can capture the traffic behaviour without being seen by road users. In addition, it has an advantage of a wide angle mode which covers a large field of view (FOV). However, this greatly increased FOV comes of the detriment of distortion being displayed into the resultant surface: this phenomena is known as fish-eyes effect (e.g. the site appears in a nonlinear fashion (see Figure 3.3 a)). This effect can be corrected using three types of calibrated parameters: intrinsic parameters, distortion coefficients and undistorted image multiplication. Distortion coefficients are used to adjust radial and tangential distortions. These parameters are estimated using a planer checkerboard in-house before using the camera. The result is an undistorted surface and smaller FOV as shown in Figure 3.3 b).

[6] Readers can see the following link for more details http://docs.opencv.org/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html.

(a) Origional captured image (b) Corrected image

Figure 3.3: An example of image with a fish-eyes effect and the resultant calibrated image

3.2.2 **Feature Tracking**

Features are tracked from frame to frame using a feature tracking algorithm. This algorithm was proposed by (Saunier & Sayed, 2006) as an extension of the work presented in (Beymer, McLauchlan, Coifman, & Malik, 1997). The features are tracked through each frame using the well-known Kanada-Lucas-Tomasi (KLT) interest point tracker (Tomasi & Kanade, 1991). The key characteristic of tracking is the ability of the algorithm to differentiate between moving objects and silent background features. Additional "filters" are added by (Saunier and Sayed 2006) to keep only relevant features. These filters deal with two issues: stationary features and feature tracker errors of irregular motions by setting reasonable thresholds over feature acceleration and change in direction.

In the TI tool, although feature tracking is performed automatically, it requires to tune the parameters in advance. These parameters control the quality and dynamic characteristics of the features which are defined beforehand using a tracking configuration file. There are a total 16 main parameters used in the configuration file concerning the tracking procedure. As a first step, a small portion of the video (e.g. 1000 frames) is used to calibrate tracking parameters by trial and error: the calibrated parameters are then used for the rest of the videos for the same scene. Among these parameters, only seven parameters have a considerable effect of feature tracking results as shown in Table 3.1. For practical purposes, the same parameters are used for the entire analysis. Feature

tracking is performed in unsupervised manner without any human intervention for the whole tracking process. Sample of tracked features points is shown in Figure 3.4.

Table 3.1: Range of used tracking parameters

Parameter	Used values range	Selected values
Maximum number of tracked features points at each frame	1000-1200	1000
Minimum feature quality	0.01- 0.06	0.06
Size of the search window at each pyramid level	6-8	7
Number of displacements to test minimum feature motion	2-4	3
Minimum displacement to keep features	0.005-0.01	0.01
Maximum feature acceleration	2-4	3
Maximum feature deviation	0.5-0.8	0.6

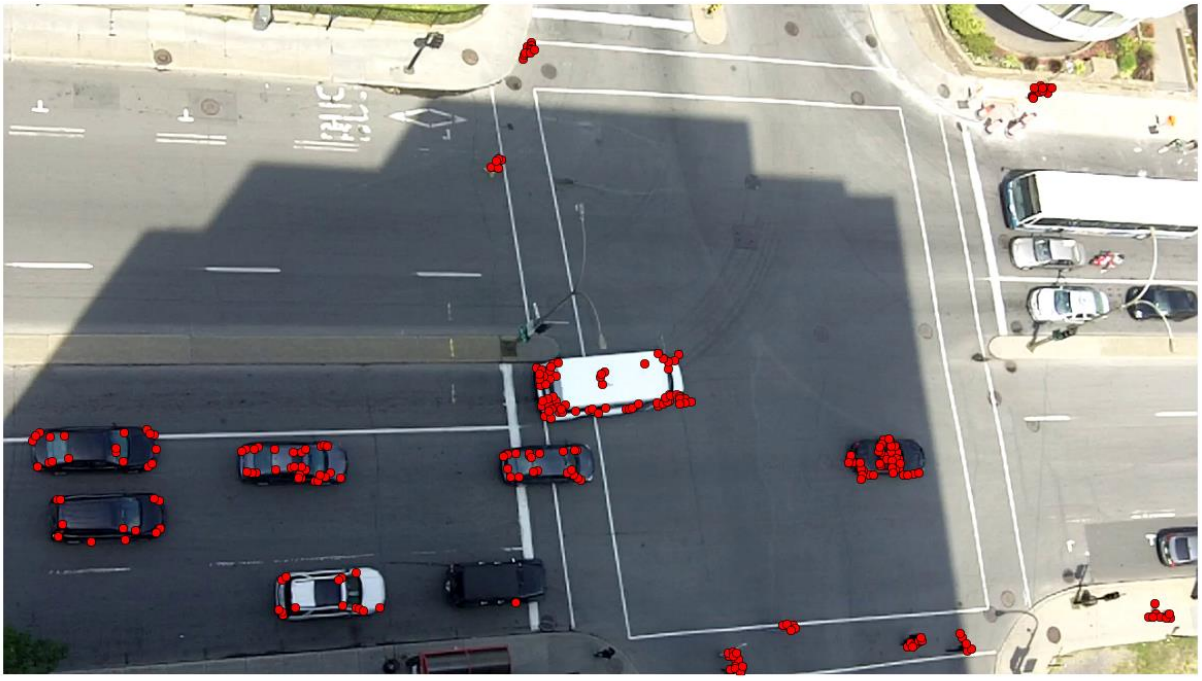


Figure 3.4: Sample of features tracked on road users

3.2.3 Feature Grouping

The output of feature tracking algorithm is a large number of trajectories. Mainly, one or more features trajectory represent one road user. The second step of video analysis is to group feature trajectories into road user (object) hypothesis: features within a defined spatial proximity that have a similar motion are grouped as an object hypothesis. In the TI tool, the grouping algorithm described by (Beymer et al,1997) was extended to handle intersections in (Saunier & Sayed, 2006). The grouping procedure is achieved using four defined parameters. These parameters are as follows:

1. The connection distance: a threshold to connect two features if the maximum distance between these features is less than connection distance.
2. The segmentation distance: a threshold to disconnect features if the difference between the maximum and the minimum distances between the two features over time exceeds this threshold.
3. The average number of features per frame over time to create an object hypothesis
4. The minimum number of frames to consider a feature for grouping

Therefore, a road user is represented by a set of feature trajectories and deriving one overall trajectory, ideally corresponding to the centroid, is not easy. The current default solution implemented in TI is the mean of the feature positions at each frame: this average trajectory is noisy and only suitable for visualization purposes, although sometimes used for analysis as well. Figure 3.5 shows an example of grouped features into vehicle and pedestrian objects (road users' classification is performed based on aggregated speed values as it will be discussed later). It is clear that the first parts of object trajectories are noisy. Tracking errors are still common and can only be partially reduced by careful tuning of the tracking parameters. These errors are also affected by different scene views and other scene characteristics. The evaluation of tracking accuracy can be performed by comparing manually annotated trajectories with the trajectories generated by the TI software. The tracking accuracy can be measured using for example the CLEAR MOT metrics and particularly the Measure Of Tracking Accuracy (MOTA) (Bernardin & Stiefelhagen, 2008). In a recent work by (Ettehadieh, Farooq, & Saunier, 2015), the tracking parameters can be optimized automatically by maximizing the MOTA. In a large scale video-based analysis of roundabouts

performed in (St-Aubin, Saunier, & Miranda-Moreno, 2015), optimization can provide improved MOTA around 85-95%, compared to around 68-74% with default parameters. However, tracking optimization is outside the research scope of this thesis.

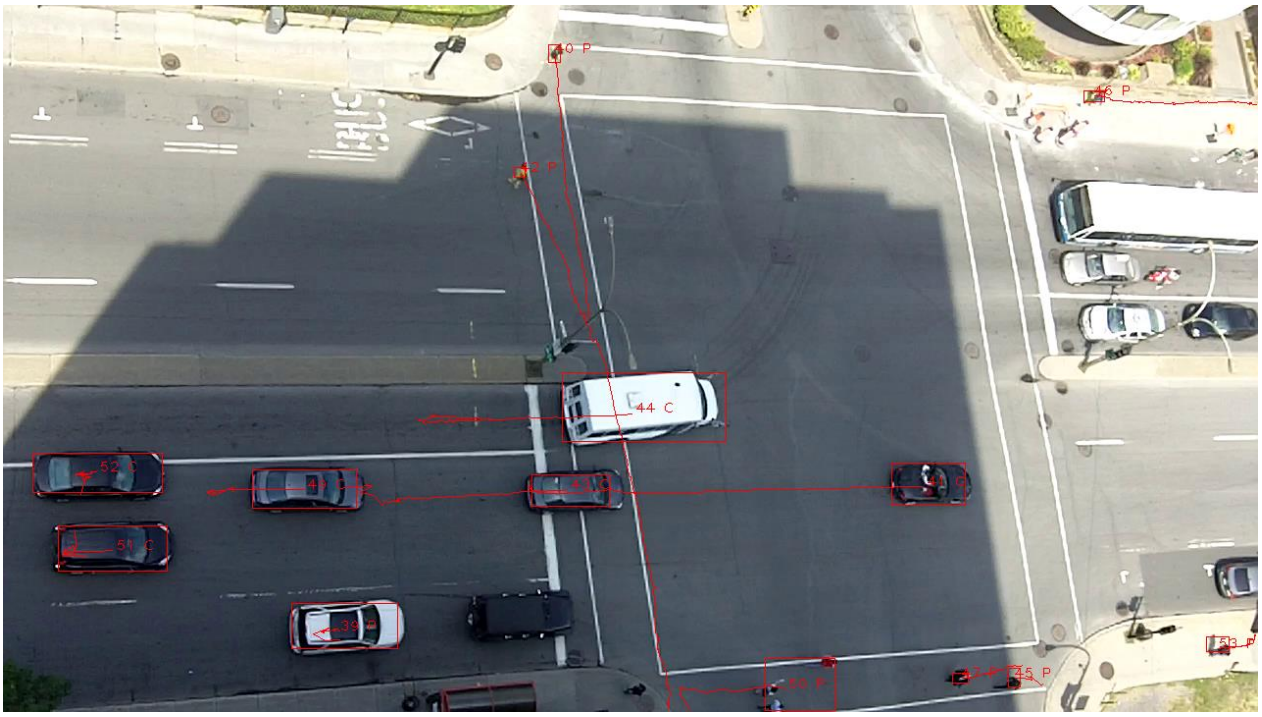


Figure 3.5: Sample of grouped features into one object hypothesis (box is the bounding box of each road user, the letter C denotes cars and the letter P denotes pedestrian)

3.2.4 New Method for Smoothing Road User Trajectories

As mentioned before, the current default road user representative trajectory implemented in TI is the mean of the feature position at each frame: this average trajectory is noisy especially when a road user enters and exits the scene and when it is partially occluded. That is why motion patterns were learnt previously from feature trajectories which are smoother (Saunier, Sayed, & Lim, 2007). Although the features track a road user well and have little noise, they have other issues such as:

- They are fragmented which affects the detection of the entry/exit zones.
- The feature trajectories constitute a larger dataset which affects the time necessary to learn the MPs.

Therefore, learning MPs based on road user trajectories would have benefits if the noise in their trajectories can be reduced.

Activity interpretation of a scene using road user trajectories derived as the mean feature positions is challenging as the noise in their positions may have an effect on the results of the clustering algorithms. A first step may be to smooth the trajectories. Standard techniques such as kernel smoothing, splines, Kalman filters are used in the literature. These techniques deal with the object positions without any prior knowledge or consideration of the kinematic characteristics of the road users. Taking into consideration that the features trajectories are almost noiseless and provide the exact kinematic characteristics of the road user, one can use those features as a prior information to smooth trajectories. Therefore, this section presents a novel smoothing method based on feature trajectories to reconstruct a smoother object trajectory: the resulting object trajectory has less noise and better reflects the road user dynamics.

Smoothing is performed in two steps: by finding 1) a single feature F_1 that is tracked during the entire existence of the object or use the longest feature and complete the missing positions from other features, 2) the parameters that represent the relationship between the feature F_1 and the mean feature position O, as shown in Figure 3.6 a), represented by the vector $\overrightarrow{F_1O}$. Since road users are rigid objects (pedestrians are somewhat less rigid), the angle θ between the feature velocity and $\overrightarrow{F_1O}$ and the distance F_1O should be constant over time. Smoothing is therefore performed by computing the median angle θ and distance F_1O over time, θ^* and F_1O^*, and computing at each instant t the new object position as the feature translated by a vector of angle θ^* with the feature velocity at t and norm F_1O^*. The same procedure is repeated for all features with a defined minimum length to avoid small features: the smoothed object coordinates are the medians of the coordinates of all corrected feature trajectories. The algorithm relies on the assumption that at least 50 % of the mean feature positions are correctly estimated by the tracker. The estimated parameters should be accurate enough to resample the noisy positions.

The smoothing algorithm performs well as shown in the examples of smoothed trajectories in Figure 3.6 b-d. To further demonstrate the performance of the algorithm, the speed profile derived from the original average object trajectory (SP1) and from the smoothed one (SP2), along with the speed derived from the average feature velocities provided by TI (SP3), are shown in Figure 3.6 e,f: SP1 is very noisy, while SP2 is almost as smooth as SP3, which is however already computed and therefore used in the next steps. The effectiveness of the proposed method is also measured quantitatively despite considering the unavailability of a ground truth dataset.

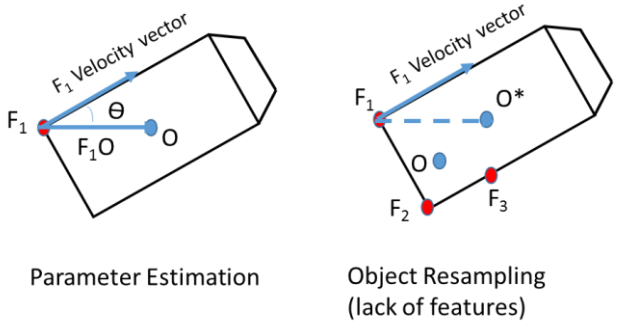
A simple quantitative measurement consists in counting the number of peaks in the speed profiles, known as a peak metric. The peak metric has been used to quantify smoothness of movements in medicine (Fetters & Todd, 1987; Kahn, Zygman, Rymer, & Reinkensmeyer, 2001; Rohrer, et al., 2002). A smooth movement is one that has fewer peaks for the entire motion. The challenge of this metric is to define the speed peaks to be counted. Kahn et al (2001) define a speed peak as any instant with a difference between two adjacent speed values exceeding a defined threshold. Defining such a threshold is a challenging and arbitrary task. In addition, it cannot differentiate between large and small peaks.

Another alternative to quantify the smoothness is based on the jerk, the third time derivative of position. Smoothness has been measured based on jerk in the literature in several forms (Hogan & Sternad, 2009): integrated squared jerk, mean squared jerk, cumulative squared jerk, root mean squared jerk, mean squared jerk normalized by peak speed, integrated absolute jerk, and mean absolute jerk normalized by peak speed. In trajectory planning for robotics manipulator, minimum-absolute jerk is used to perform the smooth motion of a robotic manipulator with few vibrations (Huang, Xu, & Liang, 2006). Cumulative Squared Jerk (CSJ) was proposed by (Flash & Hogans, 1985). In (Flash & Hogans, 1985), some experimental biomechanical studies were performed to move a hand smoothly from one point to the other. The authors concluded that the minimum CSJ can be used as a cost function to represent the smoothest trajectory among others. While previous works used jerk as a cost function for the optimization of trajectory planning, the same concept can be used for comparing different trajectories.

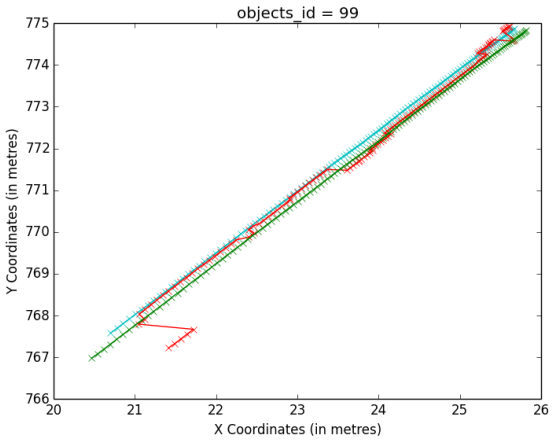
The CSJ concept is adopted to quantify the effectiveness of the smoothing algorithm. The CSJ can be calculated from the positions or the velocities using the following equation:

$$CSJ = \begin{cases} \sum_{t_f}^{t_l} (\dddot{x}(t)^2 + \dddot{y}(t)^2) \;\; if\ using\ positions \\ \sum_{t_f}^{t_l} \left(\ddot{v}_x(t)^2 + \ddot{v}_y(t)^2\right) \;\; if\ using\ velocities \end{cases}$$

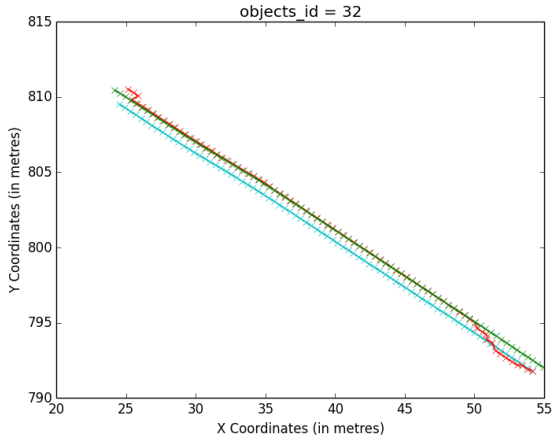
Where $\dddot{x}$ and $\dddot{y}$ are the third time derivatives of the x and y positions respectively, $\ddot{v}_x$and $\ddot{v}_y$ are the second time derivative of the v_x and v_y velocities respectively and t_f and t_l are the trajectory first and last instants.



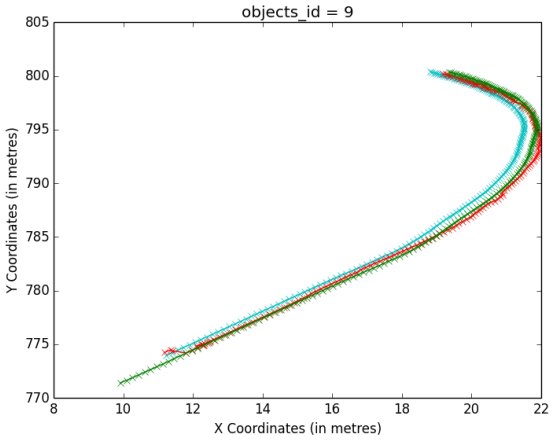
(a) object and one feature relationship



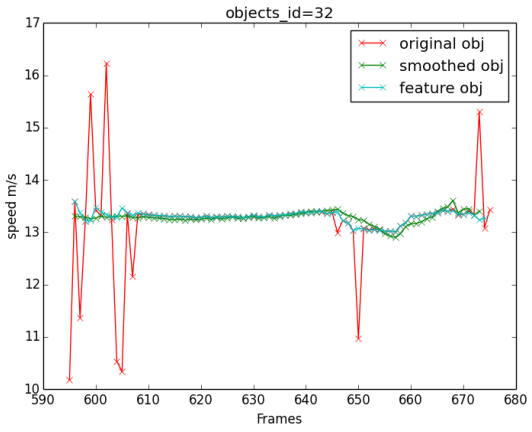
(b) an overall noisy trajectory



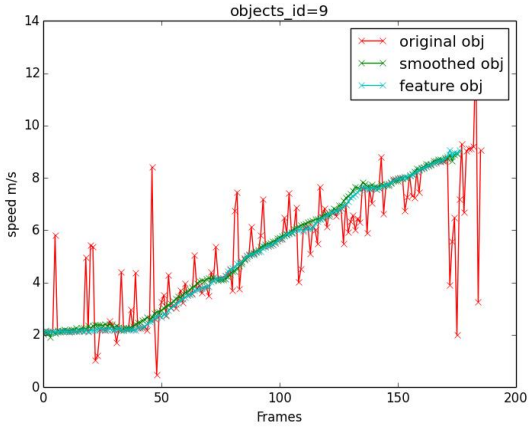
(c) noise at the beginning and end



(d) a turning vehicle



(e) derived speed profile



(f) derived speed profile

Figure 3.6: Illustration of the smoothing algorithm: a) relationship between a road user (vehicle) and one of its features, (b-d) examples of smoothed road user trajectories (in green), with the corresponding original trajectories (in red) and the selected feature trajectory (in blue), (e,f) examples of derived speed profiles.

The CSJ is calculated for any trajectory using the position information of both the original average object trajectory (TR1) and the smoothed one (TR2), and using the velocity information of the average feature velocities provided by TI (SP3). For the two road users cases shown in Figure 3.6 e,f, the CSJ values for TR2 (0.147 and 0.008) are much smaller than for TR1 (4.56 and 6.17) which confirms the effectiveness of the smoothing algorithm. Comparing the smoothness of TR2 and SP3, we found that SP3 is little smoother than TR2.

3.3 Scene Interpretation

In this section, the multi-level motion pattern learning sub-framework is described. The smoothed trajectories obtained by video tracking presented in the previous section are used as the main input for the framework. The multi-level sub-framework is summarized in Figure 3.7. Road user behavior is learnt through two main models based on the road user trajectories:

1. **Points of interest (POI)**: the POI model relies on Gaussian Mixture Models (GMM) and the Expectation Maximization (EM) algorithm. As in (Makris & Ellis, 2005), the POIs are learnt from trajectory endpoints and the results correspond to entry and exit zones as well as clusters of noisy points (e.g. caused by moving occlusions and stopping/starting vehicles). Trajectories are complete if they connect an entry zone to an exit zone, which constitutes an activity path (AP).

2. **Motion patterns (MP)**: trajectories in each AP form the training dataset for MP learning using a two-stage trajectory clustering method based on spatial and temporal information. The MPs are learnt for each AP using spatial information at the first stage, and then these MPs are further clustered using temporal information in the second stage. The hierarchical clustering is presented in Figure 3.8. Each MP is represented by its longest trajectory, known as prototype, and its associated probability is estimated using the size of each MP cluster. A cluster of small size corresponds to a low probability of occurrence and is used to detect anomalous events. Anomalies are defined as unusual behaviours in terms of position and speed (e.g. excessive speed, violations of traffic law and unsafe movement, and tracking errors). The prototypes can be used later for motion prediction for surrogate measures of safety. Motion prediction relies on matching partial trajectories to the learnt spatial prototypes to evaluate the potential for collision.

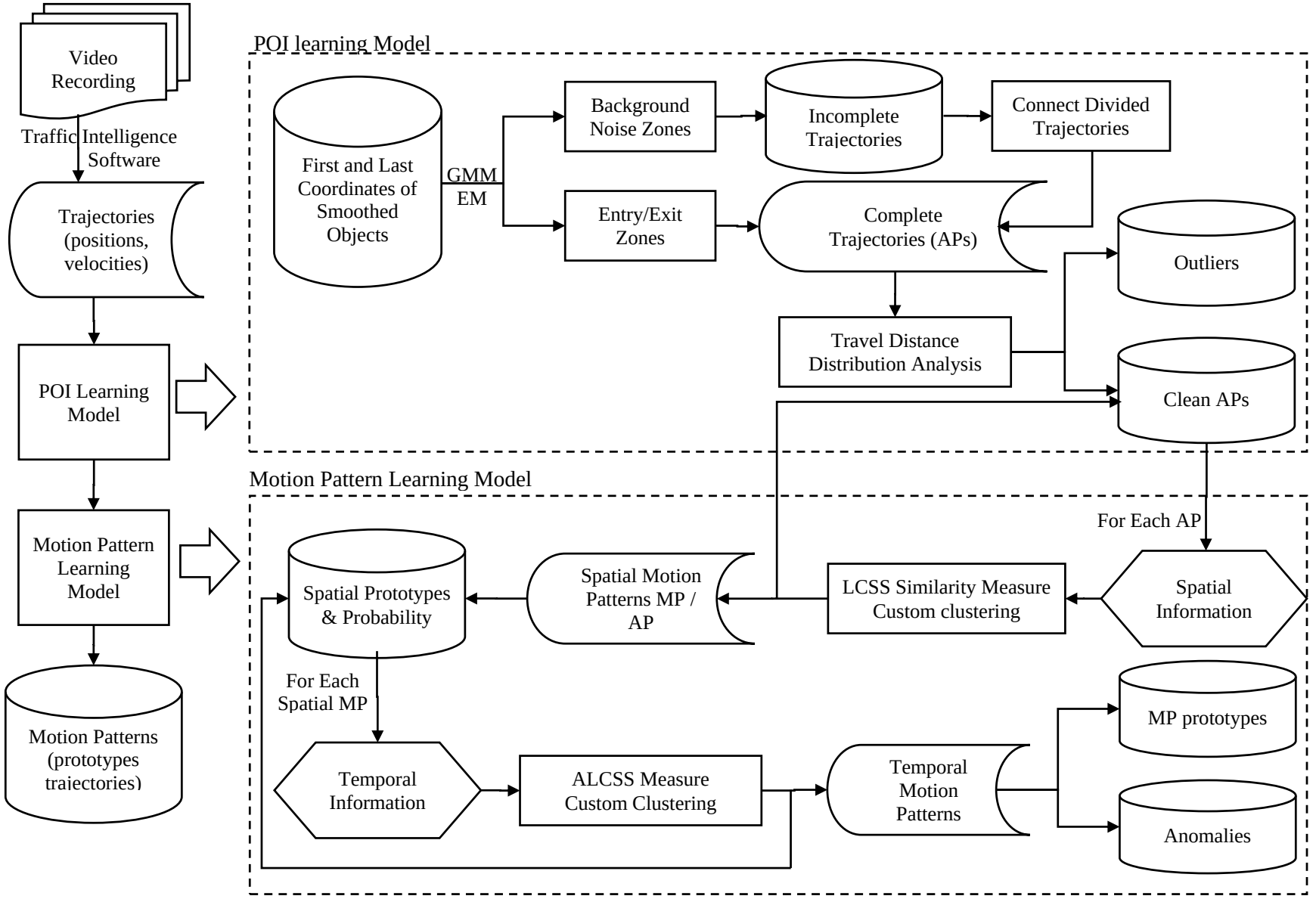


Figure 3.7: Multi-level motion pattern learning sub-framework

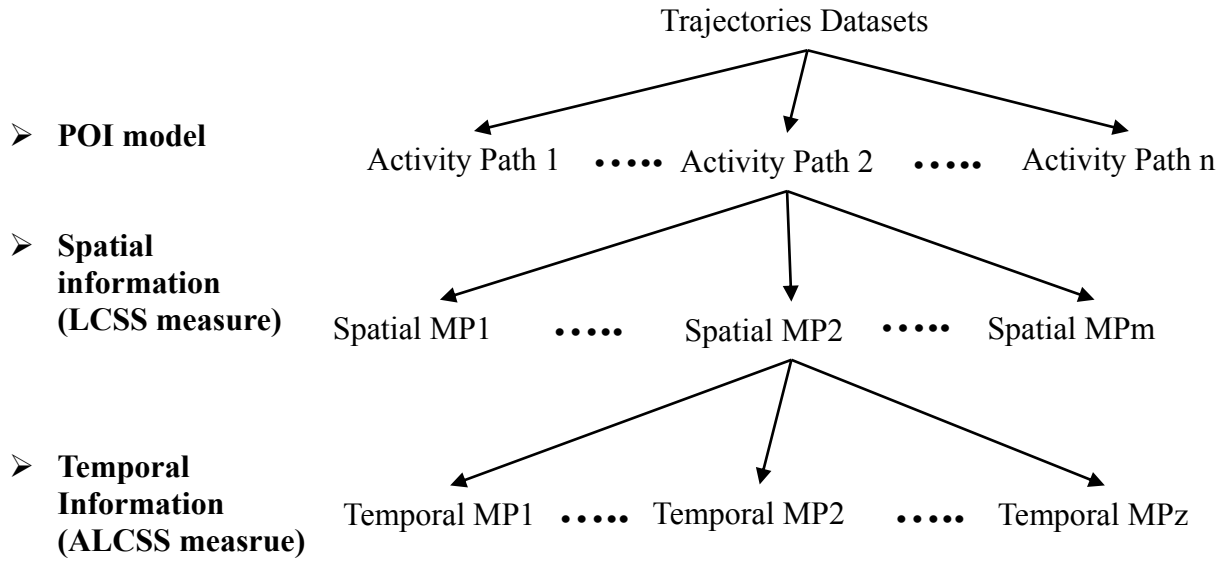


Figure 3.8: Two stage motion pattern hierarchical leaning

3.3.1 POI Model Learning

3.3.1.1 POI Detection

POIs are defined as zones where the road users appear, disappear, and/or stop for a specific moment. Traditionally, those zones are identified manually by some polygons based on a person's interpretation, which is error-prone and subjective, as well as tedious and time-consuming. Thus, identifying POIs automatically is recommended. POIs are modelled as a GMM estimated using the EM algorithm as in (Makris & Ellis, 2005). A GMM is a probabilistic model that represents all data points in form of a mixture of a finite number of Gaussian distributions. The EM algorithm is an iterative process used to estimate a GMM because it can deal with overlapped distributions. The GMM and EM algorithms are discussed in detail in Appendix A.

The different types of POIs are estimated from different datasets that are constructed specifically for that purpose. The entry and exit datasets are constituted respectively of the first and last positions of each road user trajectory. Notably, each dataset may contain points that do not correspond to actual road user entries or exits of the studied scene (that mostly happens on the border of the image), but also to areas with frequent tracking failures, caused by moving or static occlusions, i.e. occlusion by moving objects (e.g. vehicles) or static objects such as a lamp or signal post. The algorithm steps for each point dataset are the following:

1) The EM algorithm is applied to the entry or exit datasets to estimate a GMM.

2) High-density Gaussian clusters correspond to entry and exit zones (and static occlusion zones) while low-density clusters correspond to zones of moving occlusion. The clusters are classified according to their density d_i:

$$d_i = \frac{w_i}{\pi\sqrt{|\sum_i|}}$$

where w_i is the prior probability of Gaussian distribution i and $\sum_i$ is its covariance matrix. The classification threshold *Th* is defined as follow:

$$Th = \frac{\alpha}{\pi\sqrt{|\sum|}}$$

Where $\sum$ is the covariance matrix of the whole training dataset and α is a user-defined weight. The POIs that have low density, below *Th*, are classified as moving occlusion zones, often spread over most of the scene, akin to "background noise". Entry and exit zones may otherwise contain some static occlusion zones since they share the same density characteristics: in the current approach, static occlusion zones are identified manually.

3.3.1.2 POI Applications

3.3.1.2.1 Activity Paths and Trajectory Filtering

The first application of POIs is to define APs. An AP is defined by at least one trajectory going from an entry POI to an exit POI. All the trajectories that begin and end at entry and exit zones respectively are complete trajectories, while the rest are incomplete trajectories. This procedure is called the filtering algorithm.

The dataset of complete trajectories is used in the motion pattern learning. Because outliers may remain in this dataset, a pre-processing procedure is recommended. The proposed approach detects outliers by analyzing the distribution of the travelled distances in each AP: it is expected that vehicles moving along a given AP will have similar travelled distances. The extreme distances therefore represent outliers. To accomplish this, the distributions are analyzed using boxplots and their traditional statistics: the median, the first (Q1) and third (Q3) quartile, the interquartile range (IQR=Q3-Q1) and the "whisker" limits typically defined as Q1-1.5 IQR and Q3+1.5 IQR. The

usual application is to consider points outside of the whiskers as outliers. For this application, a distinction is made between mild and extreme outliers. The extreme outliers are beyond Q1-3 IQR and Q3+3 IQR which empirically correspond to grouping or smoothing errors and extreme unusual movements and are therefore removed from the AP. Mild outliers are beyond Q1-1.5 IQR and Q3+1.5 IQR, but not beyond the limits of extreme outliers. The mild outliers correspond mostly to lane changes and mild unusual movements which are kept. These outliers can be reviewed for a better understanding of the scene and activities. The procedure described above will be referred later as the cleaning algorithm.

At the end of this stage, three trajectory datasets are constituted: clean complete trajectories, incomplete trajectories, and outlier complete trajectories. The second dataset can be analyzed to connect incomplete trajectories into complete ones. The first dataset is then used in the MP algorithm.

3.3.1.2.2 Connecting Incomplete Trajectories

TI originally aimed at analyzing traffic safety in which connecting incomplete trajectories is less of a necessity. Nevertheless, studying behaviour is easier with complete trajectories and long MPs are more useful for motion prediction for surrogate safety analysis. Some fragmented trajectories may be reconstructed into complete trajectories. The connection algorithm first finds candidate incomplete trajectories using the POIs and then matches candidate trajectories using a logical connection procedure. These two steps are described in following:

1. Finding candidate incomplete trajectories using the POIs: incomplete trajectories are identified automatically in the filtering algorithm. The dataset of incomplete trajectories contains three types of trajectories: trajectories starting in an entry zone (iT1), trajectories ending in an exit zone (iT2), trajectories that do not start or end in an entry or exit zone (iT3), and occluded trajectories, defined as incomplete trajectories starting and/or ending in a static occlusion zone. The dataset is processed systematically, considering all incomplete trajectories in iT1 with incomplete trajectories in iT2 and iT3. This simple procedure is useful to identify candidate trajectories automatically and to speed up the connection of incomplete trajectories.
2. Logical connection procedure: once the candidate trajectories are identified, the connection procedure based on spatial and temporal proximity is used to connect any pair of incomplete

trajectories (T_1, T_2), where T_1 is the candidate first segment and T_2 is the candidate second segment. Spatial proximity is measured through the Euclidean distance between the last position (x_1,y_1) of T_1 and the first position (x_2,y_2) of T_2 which should be less than a defined distance Δ_1 as follows:

$$Distance = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \leq \Delta_1$$

The second constraint is the temporal proximity defined by an acceptable stopping duration Δ_2. It is measured by the difference between last instant F_1 of T_1 and first instant F_2 of T_2.

$$\text{Duration} = F_2 - F_1 \leq \Delta_2$$

Other proximity constraints can be added such as a similar motion constraint (two trajectories move in the same direction) and a pixel intensity constraint (last and first point have a similar pixel value). In case of multiple potential candidates, the trajectories with the minimum distance and minimum stopping duration are selected, with the temporal proximity having priority in the selection.

3.3.1.2.3 Efficiency Gains for MP Learning and Motion Prediction

An important challenge of learning MPs, i.e. of clustering trajectories, is to compute the similarities of all pairs of trajectories. To address that challenge, the trajectory dataset is divided into different subsets corresponding to APs. Learning the MPs for each AP separately reduces significantly the computation cost.

Regarding motion prediction, a road user enters the scene in a known POI entry zone and its partial trajectory (at each instant) needs to be matched only to the MP prototypes that share the same entry zone without the need to compute the similarity to all the MP prototypes in the scene. This simple procedure is efficient to speed up motion prediction. Besides, because the road user usually has a destination in mind, at least at the typical scale of the zones of study, one should not predict that it may leave the scene through another exit than the one he actually took. Therefore, if the trajectory has an exit zone, the partial trajectory is matched only the prototypes with the same entry and exit zones. In the last possible case where a trajectory has no POIs, the partial trajectory will be matched to the entire set of learnt prototypes, similarly to the traditional procedure (see chapter 5).

3.3.2 Two-Stage MP Learning

In the proposed approach, a slight variation of the clustering algorithm previously developed to cluster motion patterns (Saunier, Sayed, & Lim, 2007) is implemented for both stages. However, two different similarity measures are used for each stage as presented in the following sub-sections.

3.3.2.1 Spatial MP Similarity Measure

The spatial information is constituted by the position coordinates in each trajectory. With the purpose of comparing the trajectories without pre-processing that would distort the data, a similarity measure should be able to handle variable length inputs. LCSS can deal with variable length vectors and is robust to noise and outliers as some points may not be matched.

The LCSS definition is taken from (Vlachos, Kollios, & Gunopulos, 2002). Let A and B be a two trajectories with size n and m respectively, where A = {a_1, a_2….a_n}, B= {b_1, b_2,….,b_m}, and a_i and b_j are the object position coordinates. For a trajectory A, let Head (A) be the sequence {a_1, a_2,…,a_{n-1}}. Given a distance function *d* (the Euclidean distance is used) and a matching threshold ε, the LCSS between two time series A and B can be calculated iteratively as follows:

$$LCSS(A,B) = \begin{cases} 0 & if\ A\ or\ B\ is\ empty \\ 1 + LCSS\big(Head(A), Head(B)\big), & if\ d(a_n, b_m) < \epsilon \\ \max\Big(LCSS(Head(A), B), LCSS\big(A, Head(B)\big)\Big), & otherwise \end{cases}$$

In addition, a parameter δ can be added to control how far in time a point in a trajectory can be matched to a point in the other trajectory, known as trajectory bounds (positions a_i and b_j are compared only if $|i-j| \leq \delta$). Moreover, to be independent of trajectory length, the LCSS is divided by the minimum length to yield a similarity measure (SLCSS) between the two trajectories (A, B) and a distance measure (DLCSS) defined as follows:

$$SLCSS\ (A,B) = \frac{LCSS(A,B)}{\min(n,m)}$$

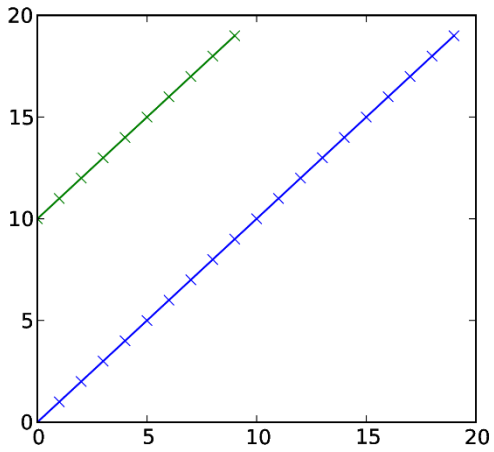
$$DLCSS\ (A,B) = 1 - SLCSS\ (A,B)$$

3.3.2.2 Temporal MP Similarity Measure

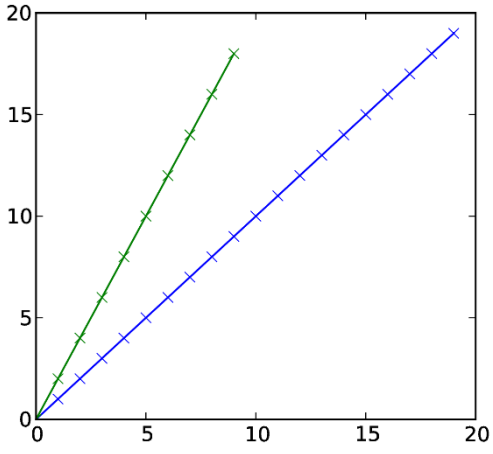
The first stage of motion pattern learning makes use only of spatial information. Temporal dynamics, measured in particular by speed, are important motion characteristics that may vary

within a spatial activity path represented by the same prototype trajectory learnt in the first stage. Therefore, the speed profile should be studied for each cluster generated by the first stage. To differentiate properly speed profiles, a similarity measure that considers the rate of change of profiles is needed. The Aligned Longest Common Sub-Sequence (ALCSS) proposed in (Saunier & Mohamed, 2014) is used for this stage.

ALCSS was developed after the observation that the existing formulations of the LCSS, with or without δ, are insufficient to measure the similarity of series if the series are simply shifted with respect to each other. Taking for example two series A=[0,1,…19] and B=[10,11…19], B is an exact sub-sequence of A (see Figure 3.9 a). LCSS(A,B) with (δ = 5) equals zero (no similarity between A and B). A possible solution is to use the simple LCSS without δ, hence LCSS (A,B) equals 1 (maximum similarity). This causes other issues as it allows any value to match any other value irrespective of the rate of change in the series (however, the order in the series is always respected). As an example of this issue, let us consider A=[0,1,…19] and B=[0,2…18] (see Figure 3.9 b): B increases at twice the rate of A and it is a sub-sequence of A. The two series should be considered different if considering the rate of change. Computing LCSS without δ results in maximum similarity which it is inappropriate. To compare the speed profiles, the rate of change should be taken into account.



a) an exact sub-sequence example



b) a rate of change series sample

Figure 3.9: Examples of simple series that illustrate the advantages of using a finite δ and aligned longest common sub-sequence (from (Saunier & Mohamed, 2014)).

ALCSS finds the best alignment of two series while allowing taking into account the rates of change by defining a finite δ. The ALCSS is computed by simply shifting the two series with

respect to each other, i.e. by adding an integer parameter *shift* to the *LCSS* computation (replacing the condition $|i-j| \leq \delta$ by $|i-shift-j| \leq \delta$) and taking the maximum *LCSS* for all possible *shift* values. The corresponding aligned similarity measure SALCSS and distance DALCSS are defined similarly to SLCSS and DLCSS.

3.3.2.3 The Clustering Method

The algorithm used for spatial and temporal clustering is the same as proposed in (Saunier & Mohamed, 2014) and is a slight variation of the algorithm previously developed to cluster motion patterns (Saunier, Sayed, & Lim, 2007). Saunier et. al. (2007) proposed a custom algorithm to learn motion patterns. This algorithm trades the parameter of the number of clusters for a maximum distance or minimum similarity between instances of the same cluster: when a new instance is too different from the existing clusters, a new one is created for it. The authors also proposed to use the original trajectories as representatives, or prototypes, in particular to use them for motion prediction. The prototypes can provide a visual and more interpretable representation of each cluster. The problem of prototypes initialization still exists in the algorithm proposed by (Saunier, Sayed, & Lim, 2007) and is a well-known challenge of many clustering algorithms such as k-means (with the initialization of the cluster centroids) as well.

Saunier and Mohamed (2014) solved the problem of dependency of the results to the algorithm initialization by favouring "long" prototypes. This is conducted in two steps as follows:

1) The elements to be clustered (trajectories or speed profiles) are sorted according to their length. In our analysis, longest cumulative travelled distance is used in case of spatial clustering while long time periods (instances) of observation is used in case of temporal clustering. Here, the longest element is considered as the first prototype.
2) For each element in the sorted elements, if its maximum similarity with all existing prototypes at the current stage is lower than a threshold, the element will be added as a new prototype. Otherwise, the element is assigned to the most similar prototype. The algorithm threshold (parameter) is the minimum similarity for two elements to be in the same cluster.

Unlike other clustering algorithms, it is not necessary to compute the similarities between all elements: it is only necessary for each element to compute its similarity to all existing prototypes at the current stage of the algorithm. For example, when a trajectory dataset contains N trajectories,

a complete similarity matrix requires $\frac{(N^2-N)}{2}$ entries using traditional computation. If the number of prototypes is m, the maximum number of necessary computations is obtained from the following equation if the first m elements are identified as prototypes. Otherwise the number of necessary computations will decrease according to when prototypes are identified.

$$maximum\ number\ of\ necessary\ computations = (N-m) * m + \frac{(m-1)(m)}{2}$$
$$= \frac{m(2N-m-1)}{2}$$

In this thesis, a slight variation of the existing algorithm is proposed to avoid clusters with few assigned elements. A minimum cluster size is used: if a cluster contains fewer elements than the defined minimum size, its prototype is removed from the set of prototypes and the associated trajectories are assigned to the most similar prototype provided that their similarities with the prototype is more than the minimum similarity. This step is called the prototype refinement algorithm.

3.4 Summary

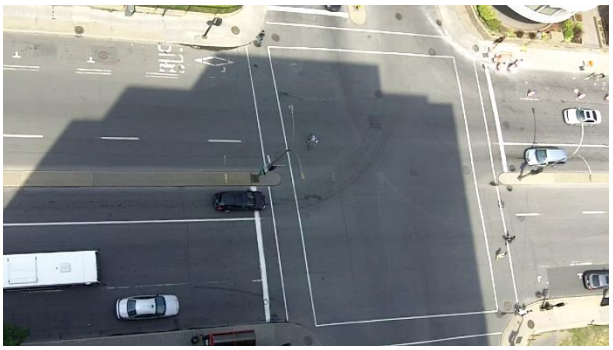
The chapter describes the proposed methodology for behaviour analysis using video data. Computer vision technique is used to extract trajectories from video recordings. A smoothing algorithm is proposed to smooth the extracted trajectories and its performance is evaluated using a quantitative criterion, called CSJ. A multi-level MP learning sub-framework is proposed for scene interpretation and behaviour analysis. POIs are learnt using the GMM and EM algorithms. The detected POIs or zones are used to remove outliers, connect fragmented trajectories, and divide the dataset into sub-datasets to speed up the MP learning phase. MPs are learnt using a two-stage procedure. The first stage clusters similar motion based on spatial information using the LCSS similarity measure and custom clustering. In the second stage, trajectories belonging to the same spatial cluster are further clustered using the temporal information. This reflects the drivers' behaviour in terms of speed profiles. A novel similarity measure (ALCSS) and the custom clustering algorithm are used to learn speed profiles. ALCSS was developed to consider the rate of change of time series. This framework can interpret traffic environment in any scene into typical motions. The learnt MPs can be used to detect anomalous activities and predict future behaviour.

CHAPTER 4 EXPERIMENTAL RESULTS (PHASE ONE)

This chapter presents the experimental results of the multi-level motion pattern learning sub-framework. It demonstrates the applicability of the proposed methodology for behaviour analysis in an unsupervised manner and highlights its usefulness and applications. The methodology is applied at three different sites. The sub-framework algorithms are implemented in the open source Python language using several scientific libraries, in particular scikit-learn (available at http://scikit-learn.org/stable/index.html) for the GMM and EM, and most are or will be available in the TI project.

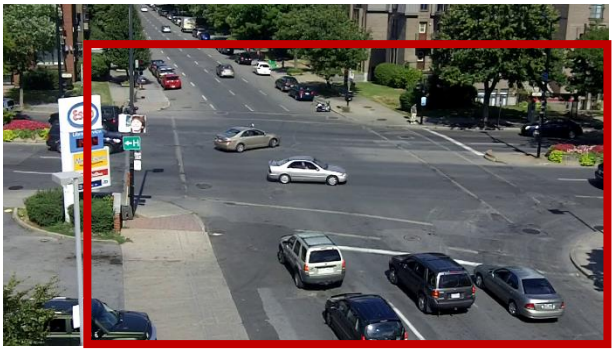
4.1 Data Description

The proposed methodology is evaluated using three different cases studies of video recordings. These case studies are intersections located in Montreal, Canada; 1) intersection of Guy Street and Boulevard Rene Levesque in downtown Montreal (Guy intersection), 2) intersection of St-Marc Street and Boulevard de Maisonneuve west in downtown Montreal (St Marc intersection) and 3) intersection of Atwater Street and Saint Jacques Street in south-west Montreal (Atwater intersection). Video data is captured using a consumer camera at a resolution of 1280 x 720 pixels (at 29.97 fps (frame per second)). Video data for Guy and St Marc intersections were recorded from a high rise building facing the intersection, while in the Atwater intersection the camera was put in the available three story building across the intersection. It is expected that the latter intersection suffers from poorer tracking and grouping caused by the low camera angle. The duration of the recorded video in each case is approximately 1 hour. The data is recorded on weekdays in August 2012 for the three sites and respectively at noon (12 p.m.), in the afternoon (3:00 p.m.), and the early evening (7 p.m.) for the Guy, Atwater, and St Marc intersections. Figure 4.1 shows the camera FOV of each intersection. It can be seen that the Guy and St Marc intersections are captured from an almost overhead view of the intersection, while the Atwater intersection has a low angle view that can lead to a high level of occlusions.



a) Guy intersection

b) St Marc intersection

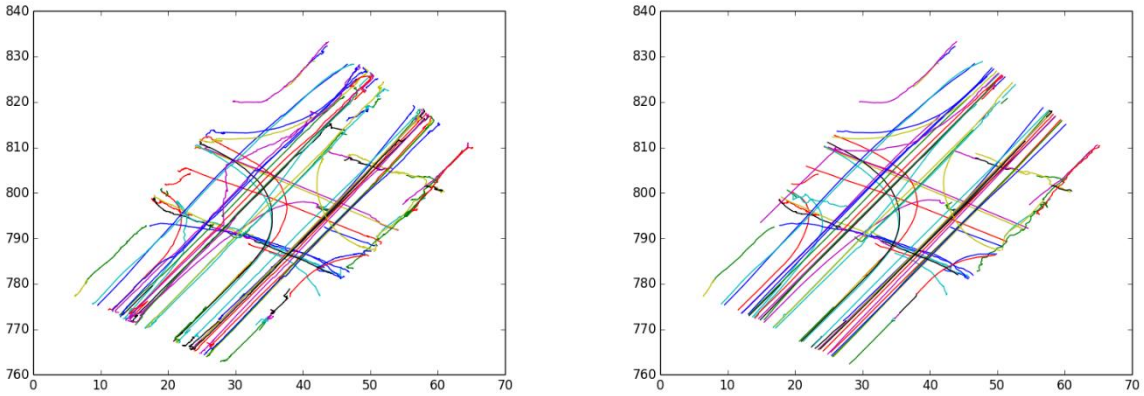


c) Atwater intersection (red box is the studied area)

Figure 4.1: Camera views for each studied intersection

4.2 Smoothing Algorithm Results

We have implemented the smoothing algorithm with the Python programming language. In order to verify our algorithm, we propose a quantitative criterion named Cumulative Squared Jerk (CSJ). A sample of the first 100 trajectories is selected randomly from Guy intersection dataset and examined as shown in Figure 4.2. This sample contains different road user trajectories (e.g pedestrians, cyclists, and vehicles). Examining performance of our algorithm visually shows its ability to remove noise effectively. This finding is confirmed quantitatively using the proposed criterion: the CSJ value for the original trajectories equals 11.394 while it equals 0.97 for smoothed trajectories. A smaller value of CSJ indicates less noise and smoother trajectories. The main limitation of this algorithm appears to be for trajectories having an over-grouping problem. Over grouping occurs when multiple (two or more) objects are tracked as one object.



a) Original trajectories b) smoothed trajectories

Figure 4.2: Smoothed and original 100 trajectories

To further demonstrate the performance of our algorithm for the three studied intersections, CSJ is computed using position information for TR1 and TR2 and using velocity information for SP3.The calculation is performed for the following datasets: the whole dataset, the vehicle trajectories only, and the pedestrian trajectories only. Road users are classified based on a simple speed criterion. The traditional speed classifier uses the maximum speed reached by the road users to distinguish between pedestrians and vehicles. As an alternative, we used 95 percentile instead of maximum function to avoid the effect of outliers. It should be clarified that cyclist trajectories could be classified as pedestrians or vehicles. Hence, a better classification is required in future investigations. The threshold used for classifier is 15 km/h.

Table 4.1 shows the results of CSJ calculation applied on all trajectories for each dataset at each site. At first glance, the mean CSJ and its standard deviation (s.d.) for the Atwater intersection are large which reflects the level of noise compared to other intersections. This more noisy data, as stated earlier, is due to the low angle of the camera which makes trajectories more prone to dynamic occlusions and over-grouping errors. Over all datasets, CSJ values for TR2 are much smaller than CSJ values for TR1 which confirms the effectiveness of the smoothing algorithm. Comparing the smoothness of TR2 and SP3, we found that SP3 was smoother than TR2. Consequently, the positions are smoothed using the proposed algorithm, while velocity datasets were chosen using the original TI computations since it was already computed and has less noise.

The results show that the algorithm performs better for vehicle trajectories than for pedestrian trajectories. CSJ can be reduced by 86-95 % for all road users, up to 97 % for vehicles, and only

up to 80-86 % for pedestrians. A possible reason of the lower performance is due to the periodic and cyclic characteristic of pedestrian motion which affect the estimation of the relationship between object and features. In addition, the assumption of constant distance and angle is violated for pedestrians, which, unlike vehicles, are non-rigid.

Table 4.1: Calculated CSJ for each dataset

Site	**Dataset (size)**	**TR1 mean (s.d.)**	**TR2 mean (s.d.)**	**SP3 mean (s.d.)**
Guy Intersection	Whole dataset (4198)	18.62 (126.69)	1.326 (16.4)	0.028 (0.105)
	Vehicle dataset(2538)	19.88 (133.10)	0.627 (8.22)	0.041 (0.133)
	Pedestrian dataset (1660)	16.706 (116.172)	2.39 (23.98)	0.006 (0.011)
St Marc Intersection	Whole dataset (3001)	6.734 (19.10)	0.96 (7.38)	0.008 (0.034)
	Vehicle dataset (941)	7.86 (12.10)	0.22 (1.27)	0.018 (0.058)
	Pedestrian dataset (2060)	6.22 (21.53)	1.29 (8.84)	0.004 (0.006)
Atwater Intersection	Whole dataset (2492)	104.81 (788.41)	4.92 (17.25)	0.097 (0.298)
	Vehicle dataset (2064)	123.25 (865.09)	5.47 (18.29)	0.114 (.325)
	Pedestrian dataset (428)	15.87 (22.98)	2.22 (10.47)	0.013 (0.056)

4.3 POI Model

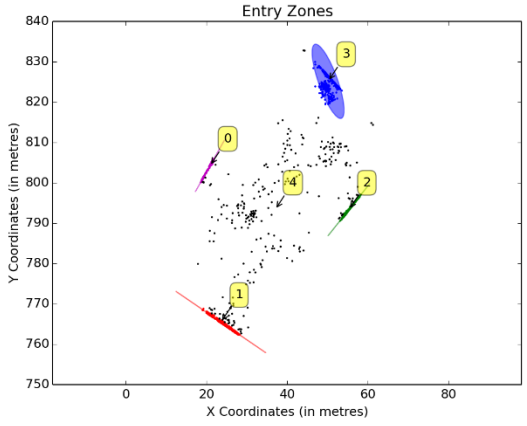
4.3.1 POI Detection Experiments

In this thesis, only vehicle trajectories were used for further analysis. The parameter for the GMM learning is the number of components or expected zones in the scene, including the noise clusters. Although component numbers can be estimated automatically using Bayesian Information Criterion (BIC), it suffers in most cases from over fitting (selecting more zones than necessary). Therefore, the number of components is chosen by trial and error for each scene. POIs are learnt and classified into entry zones, exit zones, and noise clusters based on density criterion with a defined weight α equals to 1.0. Following are the experimental results of each case study;

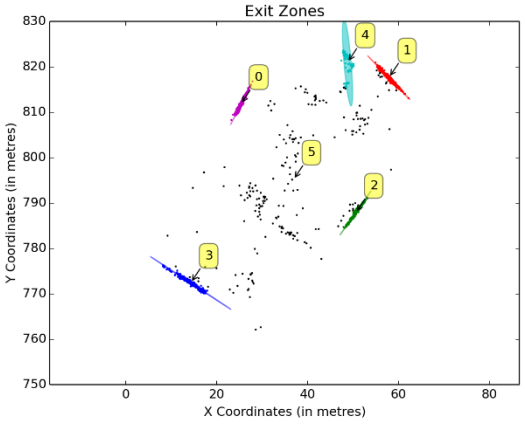
1. In Guy intersection, the number of POIs is chosen as five and six for the entry and exit datasets respectively. As shown in Figure 4.3, we were able to detect all four entry and four exit zones which can be expected as this intersection is a four legged intersection. In addition, the static occluded zone under the pole in the top-right corner of the scene is detected as an extra exit zone, but not as an additional entry zone. This is due to the closeness of the occluded zone to the entry zone: therefore it was merged with the closest entry zone producing a relatively wider Gaussian distribution. In this dataset, tracking failures caused by moving occlusion are clustered as a large Gaussian noise cluster over the whole scene. Figure 4.3 a) and b) represent the entry and exit zones including the occluded zone overlaid over a camera image. All detected POIs and their covariance ellipses are shown in Figure 4.3c,d) in world coordinates.

a) Entry Zones (image coordinates)

b) Exit Zones (image coordinates)



c) Entry Zones (world coordinates)

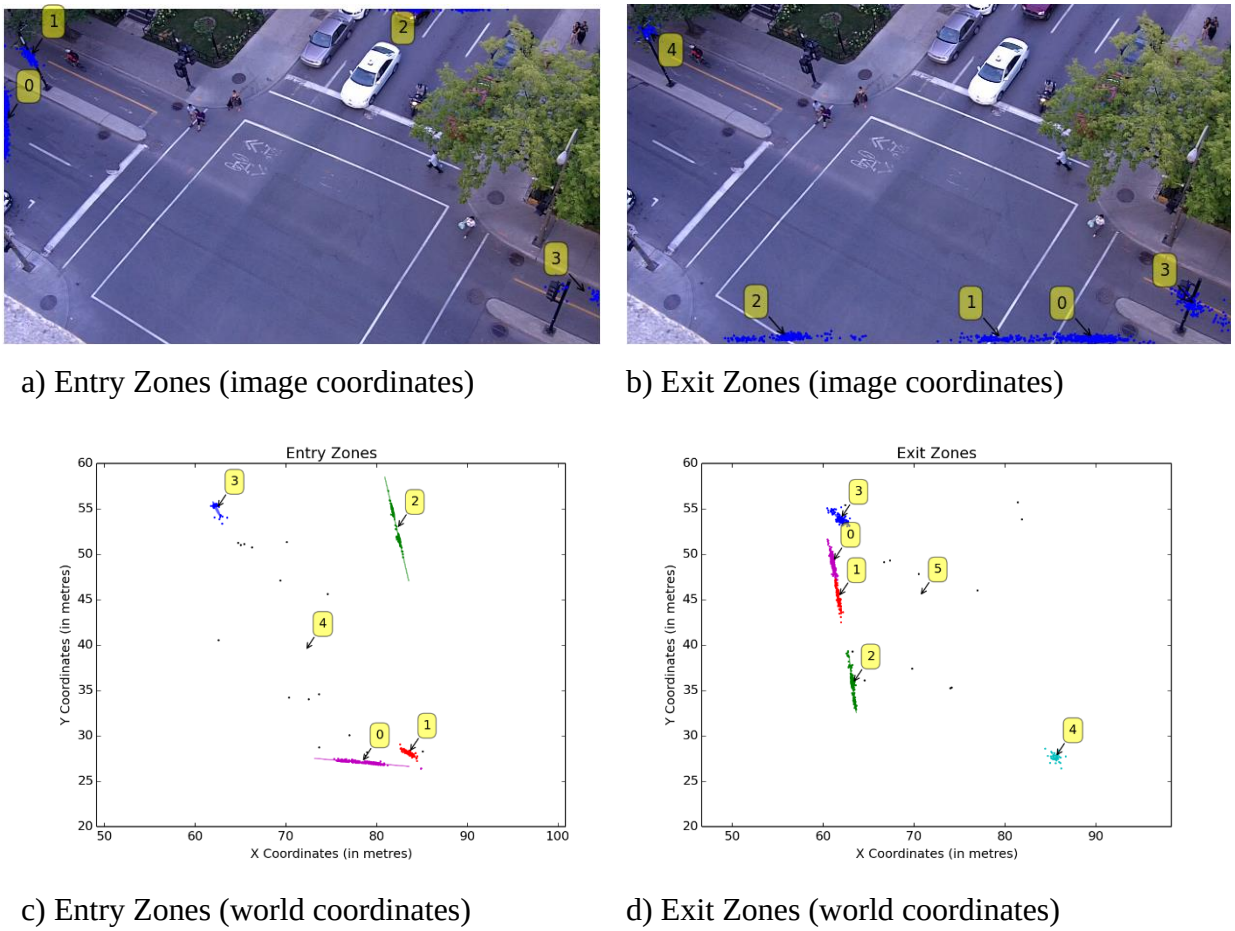


d) Exit Zones (world coordinates)

Figure 4.3: Detected POI zones for Guy intersection

2. St Marc intersection has a different layout: it is a four legged intersection with bidirectional movements in addition to a segregated bike lane. The number of POIs is

chosen as five and six for entry and exit datasets respectively by trial and error. Results shown in Figure 4.4 confirm the ability of learning all of the expected entry and exit zones. We also detected the entry and exit zones for the bike lane. The only issue noted here was the split of one exit zone into two exit zones (zones 0, 1), which may be caused by the location of the exit at the limit of the FOV which makes the exit zone longer.

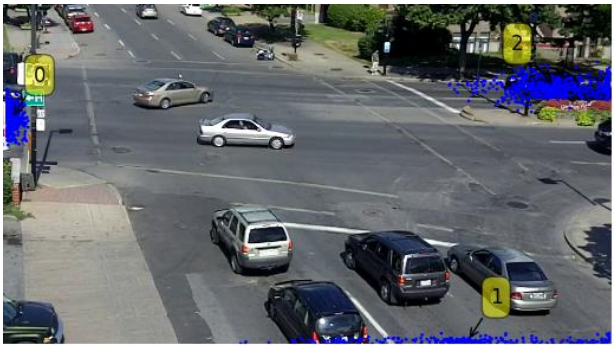


a) Entry Zones (image coordinates) b) Exit Zones (image coordinates)

c) Entry Zones (world coordinates) d) Exit Zones (world coordinates)

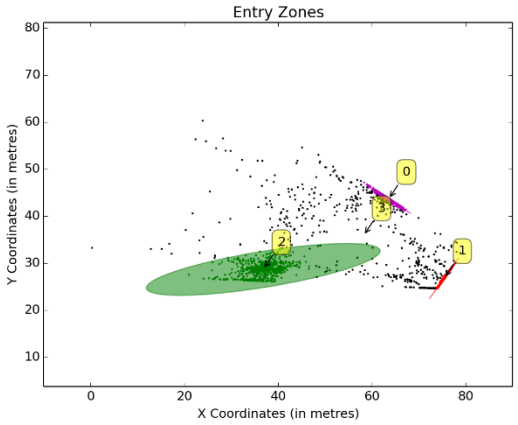
Figure 4.4: Detected POI zones for St Marc intersection

3. The Atwater intersection is a four legged intersection, where one road is unidirectional (northbound). The number of components is assumed as four for both entry and exit datasets. The results are three entry and three exit zones, plus a noise zone in each category. One of the entry zones has a wider Gaussian distribution. The reason is that this zone is located behind an area of trees and a median which leads to many fragmented trajectories. Similar to the other intersections, the beginnings and ends of noisy trajectories are represented as one cluster with a large Gaussian distribution for each entry/ exit zones. The results are summarized in Figure 4.5.

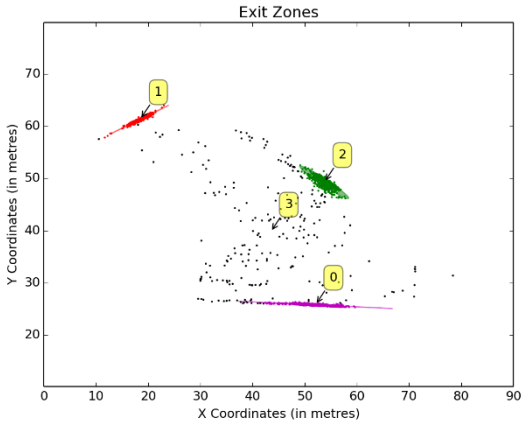


a) Entry Zones (image coordinates)

b) Exit Zones (image coordinates)



c) Entry Zones (world coordinates)



d) Exit Zones (world coordinates)

Figure 4.5: Detected POI zones for Atwater intersection

4.3.2 **POI Applications Experiments**

Once the POIs are detected, we use them for different applications to deal with trajectory noise, and to speed up computational time of the MP learning phase. This is performed with three algorithms: 1) a filtering algorithm, 2) a connection algorithm, and 3) a cleaning algorithm. The results of each algorithm are described in the following subsections for each case study.

4.3.2.1 Filtering Algorithm

Applying the filtering algorithm based on the detected POIs creates two different datasets:

a) A dataset of complete trajectories: The complete trajectories form the APs and describe the typical movements in an intersection (e.g. left turn, right turn, and through movements). This dataset will be used in the MP learning phase.

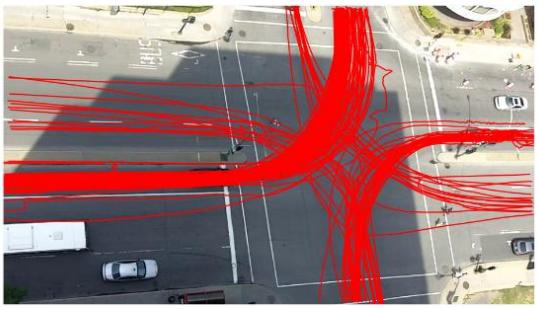
b) A dataset of incomplete trajectories: there are three types of trajectories: trajectories starting in an entry zone, trajectories ending in an exit zone, and trajectories that do not start or end in an entry or exit zone. This dataset is the main input of the connection algorithm.

Table 4.2 summarizes the results of the filtering algorithm in each case study. The Guy intersection is the only intersection with a static occlusion zone, hence an extra dataset, called occluded dataset, is found in this intersection. An occluded trajectory is defined as an incomplete trajectory starting and/or ending in a static occlusion zone. For this case study, these trajectories will not be further studies as the occlusion zone is near the border of the FOV.

Table 4.2 : Filtering algorithm results

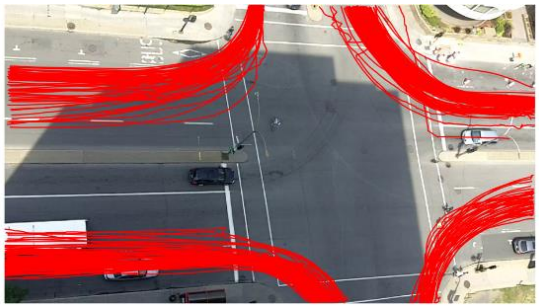
Datasets	**Intersections**		
	Guy	**St Marc**	**Atwater**
Vehicle trajectories size	2538	941	2064
Complete dataset	1312	654	1442
Incomplete dataset			
• Starting in an entry zone	423	54	125
• Ending in an exit zone	601	229	392
• Do not start and end in an entry or exit zone	88	4	105
Occluded dataset	114	0	0

The datasets of complete trajectories, corresponding to the APs and the traffic volumes of each movement are shown in Figure 4.6, Figure 4.7, and Figure 4.8 for the Guy, St Marc, and Atwater intersections respectively. The traffic volume is estimated using the number of road users classified as vehicles for each movement per one hour (the studied period). In the St Marc intersection, vehicle trajectories include misclassified cyclists trajectories. The complete trajectory dataset has five trajectories that represent cyclists moving from vehicle lanes to the cyclist lane or vice versa. In addition, a motorcyclist trajectory was detected as moving in the cyclist lane but its entry zone was detected in the vehicle entry zone. The entry and exit zones could be used to reclassify cyclists that move in the cyclist lane and are classified as pedestrians. This application is out of our research scope as we are interested in vehicle trajectories only.

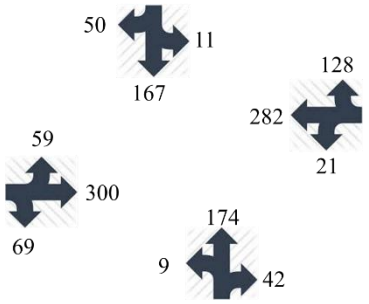


a) Complete Trajectories (APs): Left Turn

b) Complete Trajectories (APs): Through

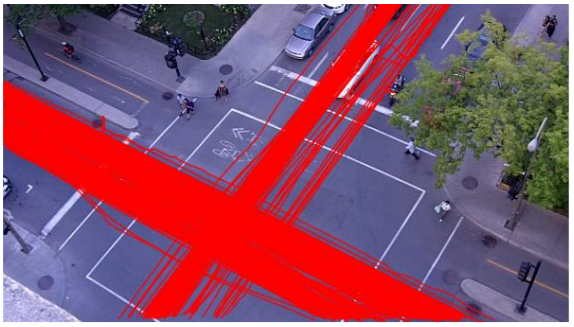


c) Complete Trajectories (APs): Right Turn



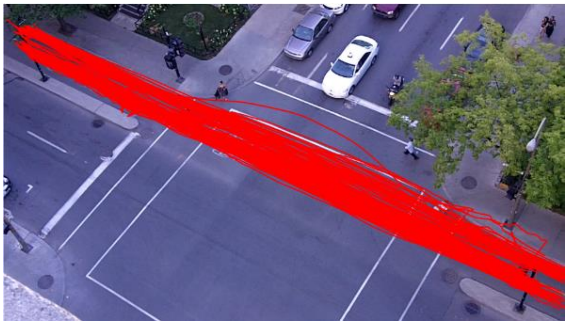
f) Traffic volumes of each AP

Figure 4.6**:** POI applications: filtering algorithm results for the Guy intersection

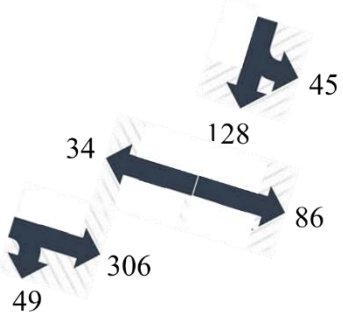


a) Complete Trajectories (APs): Through

b) Complete Trajectories (APs): Turning

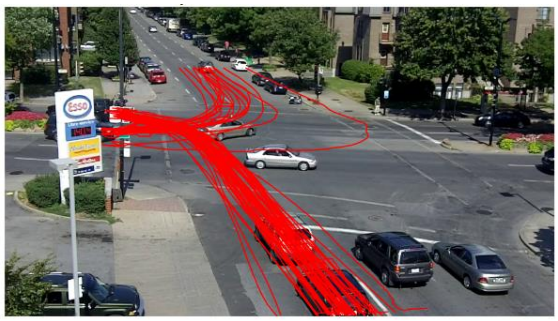


c) Complete Trajectories (APs): Cyclist

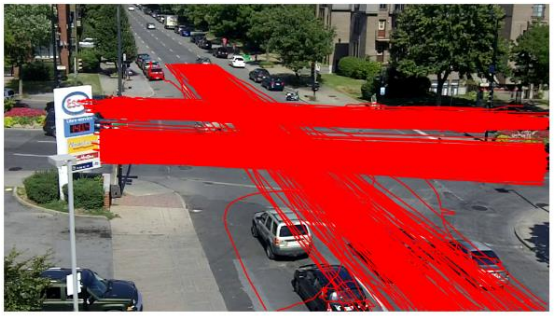


d) Traffic volumes of each AP

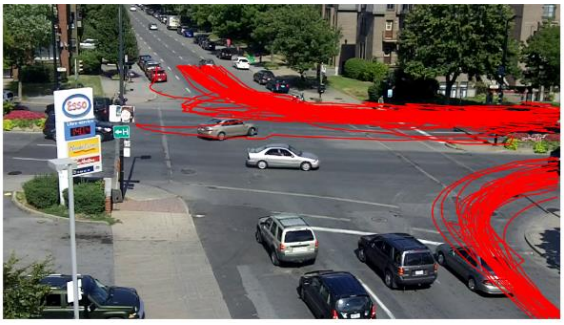
Figure 4.7**:** POI applications: filtering algorithm results of the St Marc intersection



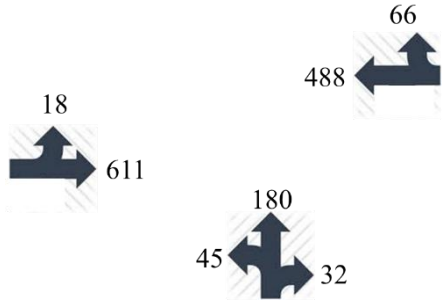
a) Complete Trajectories (APs): Left Turn



b) Complete Trajectories (APs): Through



c) Complete Trajectories (APs): Right Turn



f) Traffic volumes of each AP

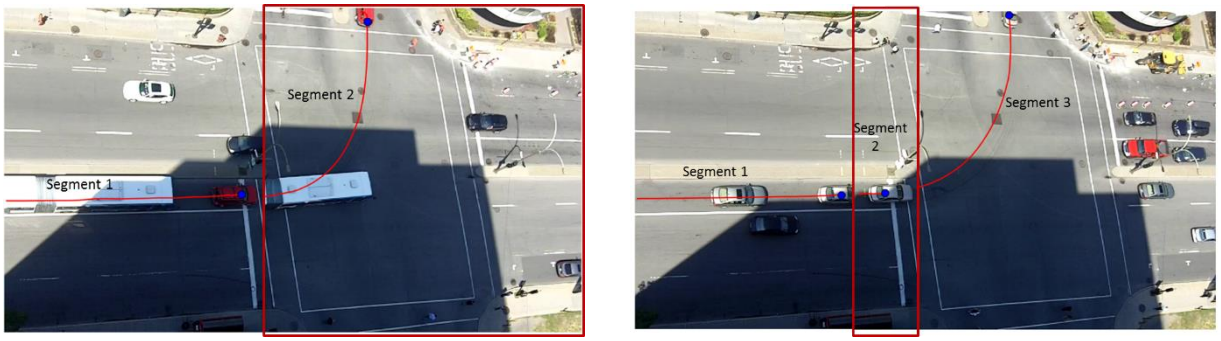
Figure 4.8**:** POI applications: filtering algorithm results of the Atwater intersection

4.3.2.2 Connection Algorithm

The second application of POI is to connect incomplete trajectories.. The thresholds for spatial and temporal proximities are chosen as 1.5 meter and 2000 frames (66.7 sec) respectively. The large value of stopping time is to cover the maximum possible waiting time due to the traffic signal phase.

The experimental results for the Guy intersection show that 464 incomplete trajectories are connected and result in 232 complete trajectories. To test the performance of the connection algorithm, we reviewed the connected trajectories manually by watching the video. In this dataset, we found 226 complete trajectories to be correctly connected, which corresponds to a connection accuracy of 97.4 %. In addition, 118 incomplete trajectories are connected, but still do not end in an exit zone. Likewise, 170 incomplete trajectories are connected, but still do not start in an entry zone. Finally, 21 complete trajectories are merged from three segments (63 incomplete trajectories). These trajectories correspond to a specific motion behaviour in an intersection: entering the scene then stopping (segment 1), moving slowly while waiting for the signal to be green or following another moving vehicle (segment 2), and then moving until the exit from the scene (segment 3). Notably, the second segment might be split into more sub-trajectories and be

mainly classified as a pedestrian because of its low speed. Figure 4.9 illustrates two examples of connected trajectories. The ultimate goal of learning normal behaviour is to use the learnt prototypes for future motion prediction. Hence, trajectories consisting of three or more segments, representing interrupted movements through the intersection, are not considered. However, two-segment connected trajectories are used and added to the dataset of complete trajectories.



a) Connected Trajectories (2 segments) b) Connected Trajectories (3 segments)

Figure 4.9: Examples of connected incomplete trajectories

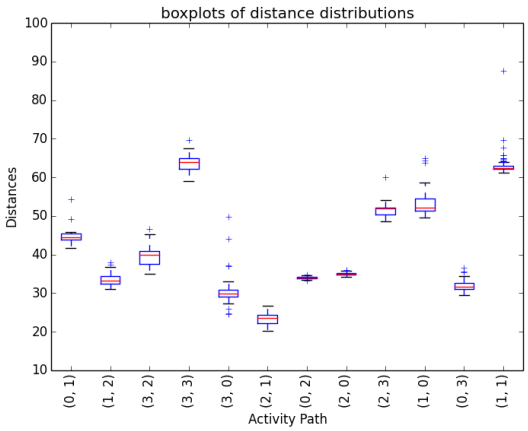
In St Marc intersection, the connection algorithm succeeded in connecting 306 incomplete trajectories. The accuracy of connected trajectories is 98.7 % with only two trajectories falsely connected. Moreover, 60 incomplete trajectories are connected but still had missing entry or exit zones. For the Atwater intersection; the connection algorithm works less accurately. Only 82 incomplete trajectories are connected for which 70 % are correctly connected. Our explanation is that the low angle FOV affects the performance of tracking and the grouping algorithm which has a direct influence on the connection algorithm. In conclusion, the detection and tracking algorithms require further investigation for the low angle FOV. However, the performance of POI for detecting incomplete trajectories and using them in the logical connection algorithm performed well overall in an unsupervised manner.

4.3.2.3 Cleaning Algorithm

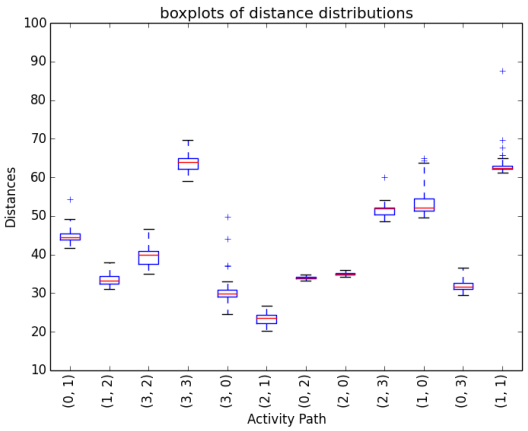
The third application of POIs is the cleaning algorithm which involves the detection of outliers from the dataset of complete trajectories. The outliers are detected based on the analysis of the distribution of travelled distances in each AP. Two types of outliers exist: extreme outliers and mild outliers. The results for the Guy intersection show that the dataset contains 23 mild outliers and 13 extreme outliers. Figure 4.10 a,b) show the boxplots of the travelled distance distribution

for both mild and extreme cases. The only difference between Figure 4.10 a) and Figure 4.10 b) is the whisper limits. Trajectories considered as mild outliers are presented in Figure 4.10 c), while others considered as extreme outliers are shown in Figure 4.10 d). Noticeably, extreme outliers are very noisy trajectories or represent an extreme abnormal movement. Figure 4.10 e,f) show some examples of mild and extreme outliers. The difference between mild and extreme outliers can be seen in unusual left turn movement examples. Although each outlier corresponds to an unusual left turn movement, the mild outliers are smoother than extreme ones.

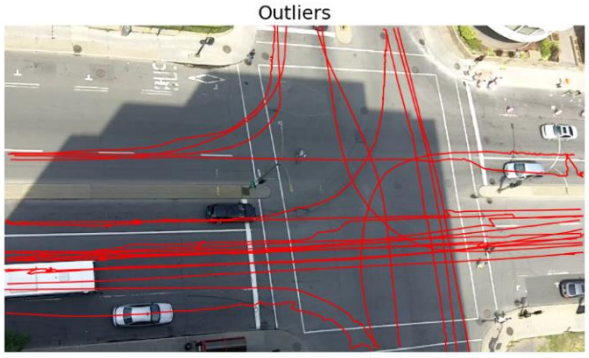
When testing the cleaning algorithm in the St Marc intersection, 20 mild outliers and 14 extreme outliers are detected. As shown in Figure 4.11, it is noticed that all extreme outliers are connected trajectories where significant noise at the connection location is detected. Mild outliers are a mix of lane change trajectories and connected trajectories. For the Atwater intersection, the dataset contains 37 mild outlier trajectories and 17 extreme outliers, all of which are due to grouping and smoothing errors. Figure 4.12 shows the detected outliers for both mild and extreme outliers, where the main difference between the types of outliers is found to be the noise levels. Therefore, it is suggested to keep the mild outliers for the motion pattern learning, and to remove the extreme outliers from the complete datasets.



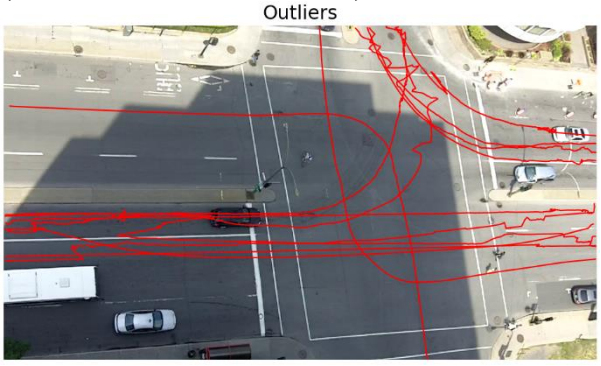
a) Boxplots of distance distribution per AP (Mild outliers ±(1.5 IQR : 3IQR))



b) Boxplots of distance distribution per AP (Extreme outliers > ±3 IQR)



c) Detected mild outliers



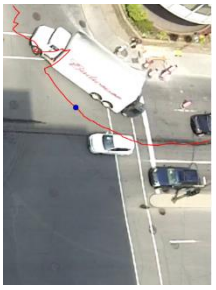
d) Detected extreme outliers

Unusual overtaking

Mild unusual left turn

e) Examples of detected mild outliers based on distance distribution

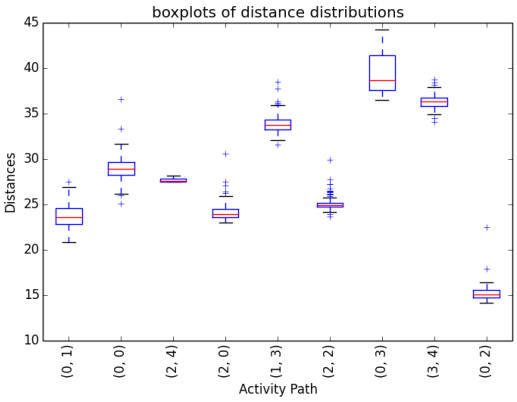
Extreme unusual left turn



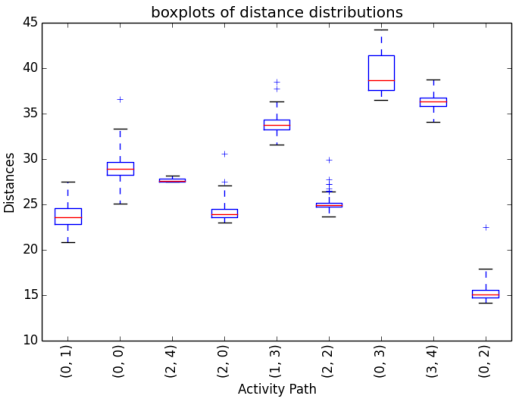
Over-grouping with smoothing errors

f) Examples of detected extreme outliers based on distance distribution

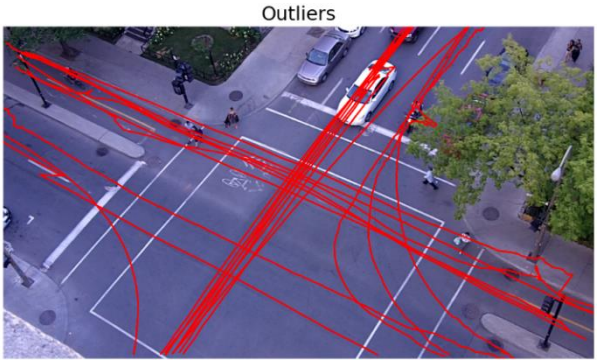
Figure 4.10: POI applications: cleaning trajectories using boxplots based on distribution of travelled distances in each AP (Guy intersection)



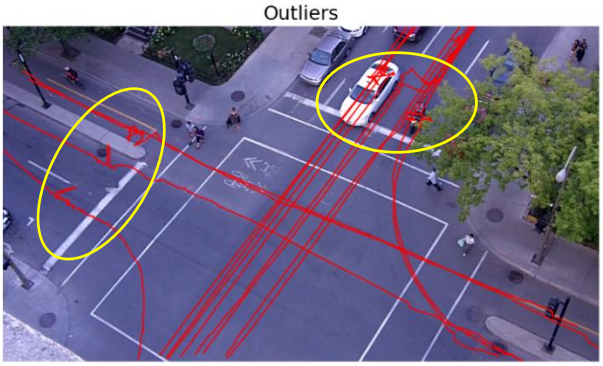
a) Boxplots of distance distribution per AP (Mild outliers ±(1.5 IQR : 3IQR))



b) Boxplots of distance distribution per AP (Extreme outliers > ±3 IQR)

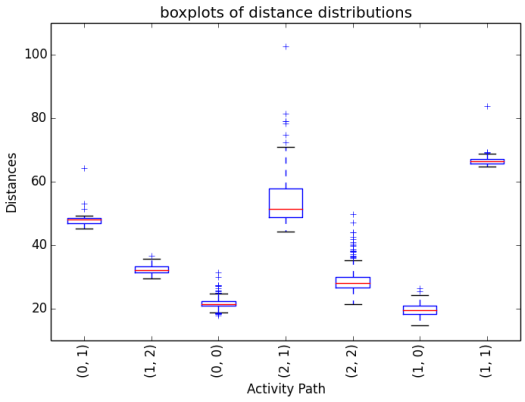


c) Detected mild outliers

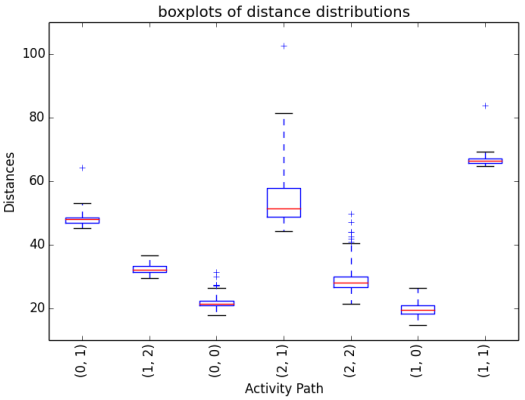


d) Detected extreme outliers (ellipses indicate the connection locations)

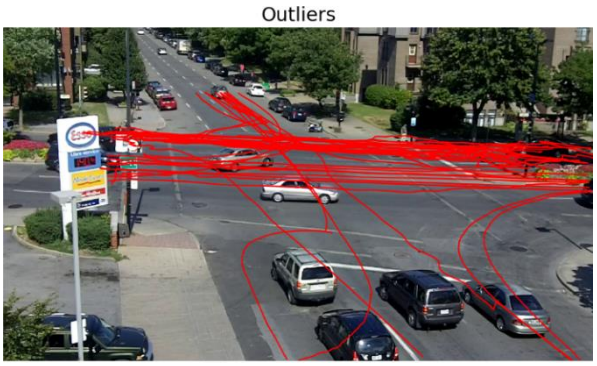
Figure 4.11: POI applications: cleaning trajectories using boxplots based on distribution of travelled distances in each AP (St Marc intersection)



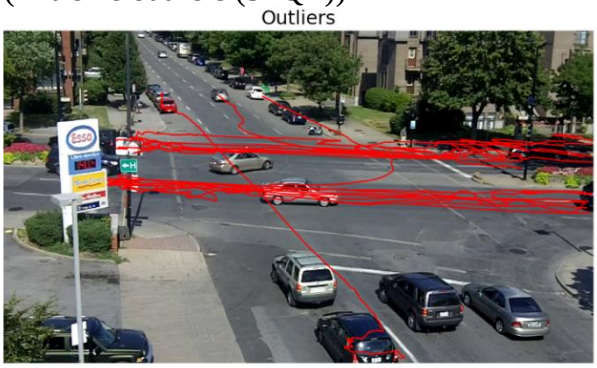
a) Boxplots of distance distribution per AP (Mild outliers ±(1.5 IQR : 3IQR))



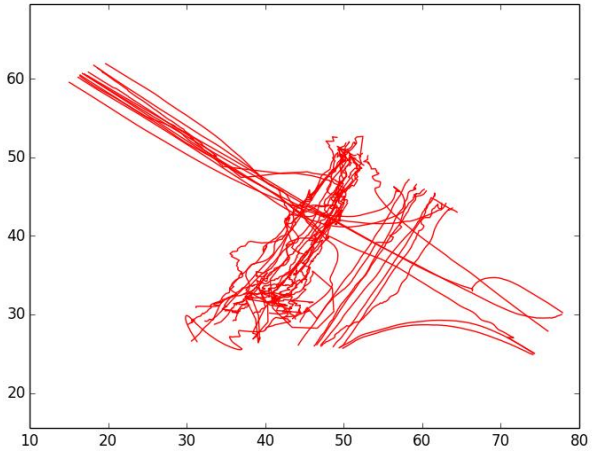
b) Boxplots of distance distribution per AP (Extreme outliers (3 IQR))



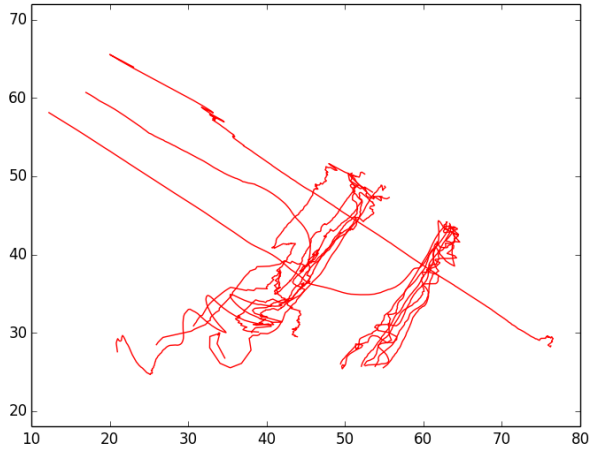
c) Detected mild outliers



d) Detected extreme outliers



c) Detected mild outliers (world coordinates)



d) Detected extreme outliers(world coordinates)

Figure 4.12: POI applications: cleaning trajectories using boxplots based on distribution of travelled distances in each AP (Atwater intersection)

4.4 Two-Stages MP Learning Experimental Results

4.4.1 Speeding up the computational cost

The main challenge of MP learning comes from the need to compute all pairwise similarities between trajectories. The construction of a pairwise similarity matrix is computationally inefficient in both computational time and storage space: the proposed framework addresses precisely this issue. To avoid unnecessary computations of trajectory similarities, two procedures are proposed and their performances are tested as follows:

1) *Using the proposed clustering method and considering only the number of necessary computations*: This procedure is tested in a sample dataset containing 167 trajectories and grouped into 13 clusters (prototypes). This dataset represents the through movement from north to south in the Guy intersection. The number of computed similarities for the traditional procedure equals 13861, while it is only 2080 in the proposed clustering method. The reduction of the number of computed similarities translates into savings of processing time. For this example, the gain is around 85 % of the overall computation time.

2) *Using POIs and dividing the trajectory dataset into different sub-datasets corresponding to APs*: learning the MPs for each AP separately reduces significantly the computational cost. To test the effectiveness of our algorithm in speeding up the computation for MP learning, a sample of 200 complete trajectories is used: their complete similarity matrix is constructed based on the traditional method without dividing the dataset first into sub-datasets (APs). The run time using the traditional method is 1600 s, while the proposed method takes only 140 s: the gain is more than 90 %. In addition, our procedure is able to reduce the required storage space. For instant, we have 1531 trajectories in Guy intersection divided into 12 APs. For each AP, the similarity matrix is computed and saved in a separate file; the total required space of all 12 files is 8.72 MB. However, the required space based on a traditional procedure is 6.6 times larger (57.54 MB). Dealing with small files is easier in importing and exporting data than dealing with a large file. This procedure does not affect the final cluster results where similar clusters are obtained in the traditional and the proposed methods. Despite the simplicity of this procedure, it allows to analyze large datasets more efficiently.

4.4.2 MP Learning Results

The MP learning algorithm uses the set of complete trajectories after removing the outliers. Table 4.3 details the size of each dataset and the number of learnt spatial and temporal prototypes for each case study. The parameters for the learning phase are the matching threshold ε and the minimum cluster similarity. These are chosen by trial and error respectively as 1.0 m and 0.75 for spatial information and 1 m/s and 0.75 for temporal information. For ALCSS, δ equals 2 frames. The learnt prototypes for the two learning stages are presented in Figure 4.13, Figure 4.14 and Figure 4.15 for the Guy, St Marc and Atwater intersections respectively. Although such unsupervised learning is difficult to evaluate, the results of MP learning based on spatial information suggest an acceptable division of the trajectories (as seen in top part of Figure 4.13, Figure 4.14 and Figure 4.15). As expected, results from the second stage based on temporal information provide more prototype trajectories (as seen in bottom part Figure 4.13, Figure 4.14 and Figure 4.15).

Table 4.3: Size of datasets and number of learnt prototypes

Dataset	**Dataset Size**	**Number of spatial prototypes**	**Number of temporal prototypes**
Guy Intersection	1531	133	260
St Marc intersection	566	28	86
Atwater intersection	1483	71	178

a) First-stage (position-based) prototypes (133 prototypes)

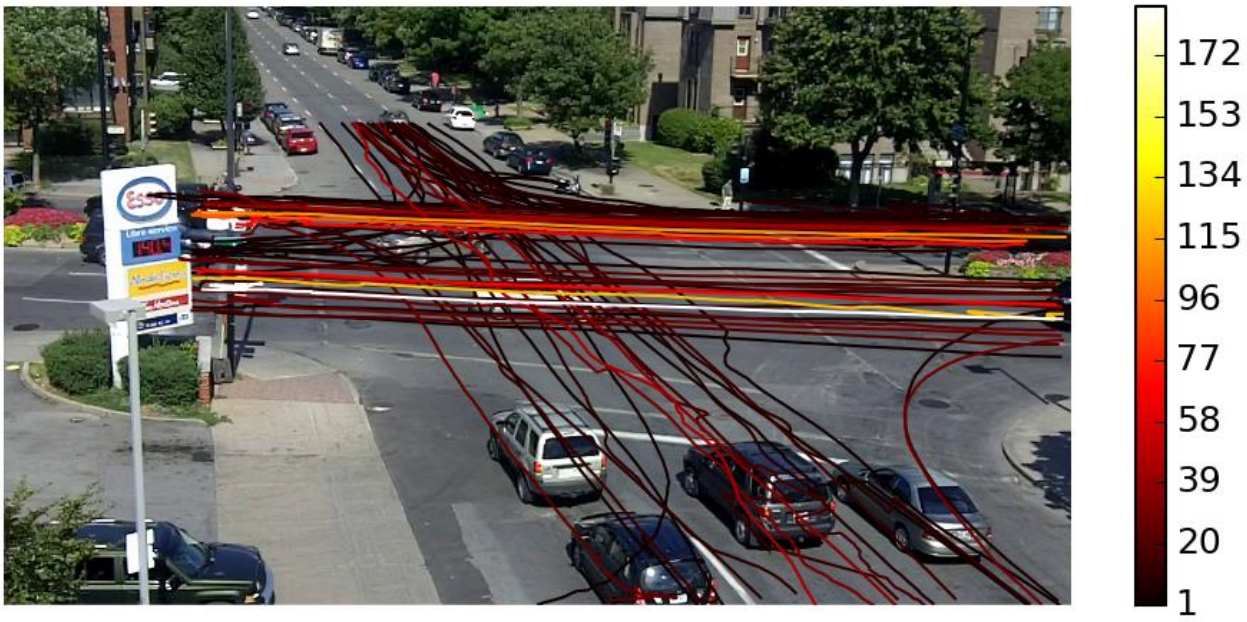
b) Second-stage (speed-based) prototypes (260 prototypes)

Figure 4.13: Prototypes representing MPs for the Guy intersection (color-coded as a sequential black-red-yellow-white to represent the number of trajectories associated to the MP)

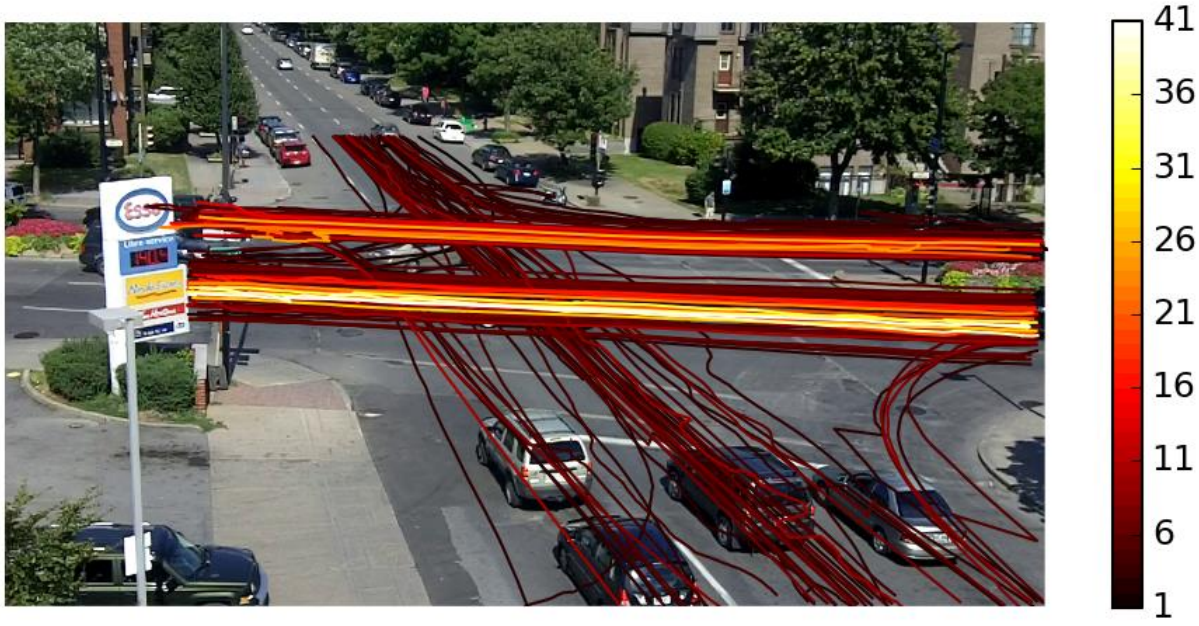
c) First-stage (position-based) prototypes (28 prototypes)

d) Second-stage (speed-based) prototypes (86 prototypes)

Figure 4.14: Prototypes representing MPs for the St Marc intersection (color-coded as a sequential black-red-yellow-white to represent the number of trajectories associated to the MP)



a) First-stage (position-based) prototypes (71 prototypes)



b) Second-stage (speed-based) prototypes (176 prototypes)

Figure 4.15: Prototypes representing MPs for the Atwater intersection (color-coded as a sequential black-red-yellow-white to represent the number of trajectories associated to the MP)

4.4.3 **Anomaly Detection**

Anomaly detection aims to discover anomalous (unusual) events defined as having a low probability of occurrence. In this thesis, the low probability can be translated into small cluster size. Hence, trajectories that are not assigned to any MP at the end of the learning phases are considered as anomalies. To distinguish between the cleaning and anomaly detection methods, the difference and relationship between both methods are discussed in the following subsection.

The cleaning algorithm aims to detect the outliers as a pre-processing procedure of MPs learning. The outliers mostly have the longer cumulative distances within each AP as previously presented. Including these outliers would lead to two issues:

1) No trajectory is assigned to this prototype which is then considered as an anomaly in the end. This leads to the unnecessary computation of similarities between all trajectories and this prototype.

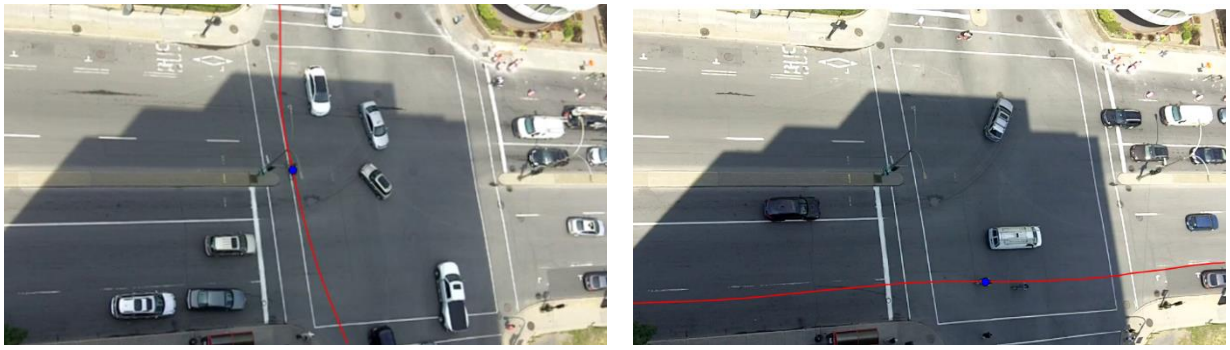
2) In some rare cases, some trajectories are similar in some parts to the outlier which may achieve the condition of maximum similarity. Therefore, this outlier will be considered as a representative of specific MP. It will influence the accuracy of predicted future motion latter.

Therefore, the objective of the cleaning algorithm is to avoid these situations and remove the outliers before the MP learning. As stated, the algorithm relies only on the cumulative travelled distance. Conversely, anomaly detection is performed by analyzing the similarities using all positions in the trajectory. Consequently, some anomalies may have a similar travelled distance but they are dissimilar to other MPs. The objective of anomaly detection is a further identification of anomalies at a higher level of analysis. For example, a normal movement that rarely happen may be detected by the anomaly detection algorithm while its travelled distance is similar to the other trajectories in its AP. Moreover, only extreme outliers are removed and mild outliers are included in the MP learning. Thus, some of the mild outliers can be further detected as anomalies. To distinguish between the detections of the cleaning and anomaly detection methods, the former are called unusual movements and the latter abnormal movements or anomalies. In addition, we should clarify that the role of these methods is limited since the detected anomalies are reviewed.

The parameter for anomaly detection is the minimum cluster size that was selected as the largest of 3 elements or 10 % of the dataset size. The dataset is the AP and the spatial MP when learning the spatial and temporal MPs respectively. It should be clear that it is difficult to detect anomalies (if they exist at all) in small datasets. However, if detected, the anomalies should be carefully reviewed manually. In our dataset, there are many sources of anomalies such as road user misclassification, tracking issues, normal but rare movements, and safety problems (e.g. violations of traffic law and unsafe movements). Examples of safety problems can be summarized as follows:

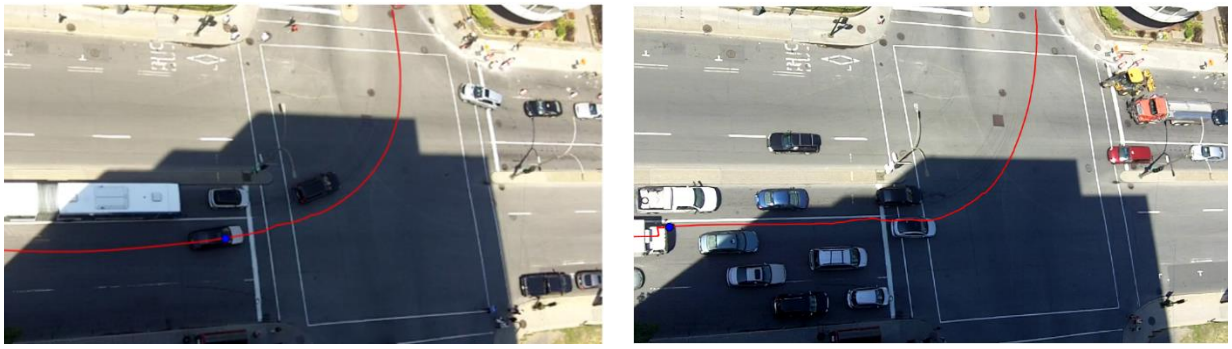
- a lane change happens by crossing a white solid line which could indicate a side-swipe interaction with or without a collision,
- improper turns: driver may only turn left or right from the left and right hand lanes unless signs or marking on the intersection indicate otherwise. Such turns are detected as anomalies,
- excessive speed.

Samples of spatial and temporal anomalies are presented in Figure 4.16 and Figure 4.17 for the Guy intersection.



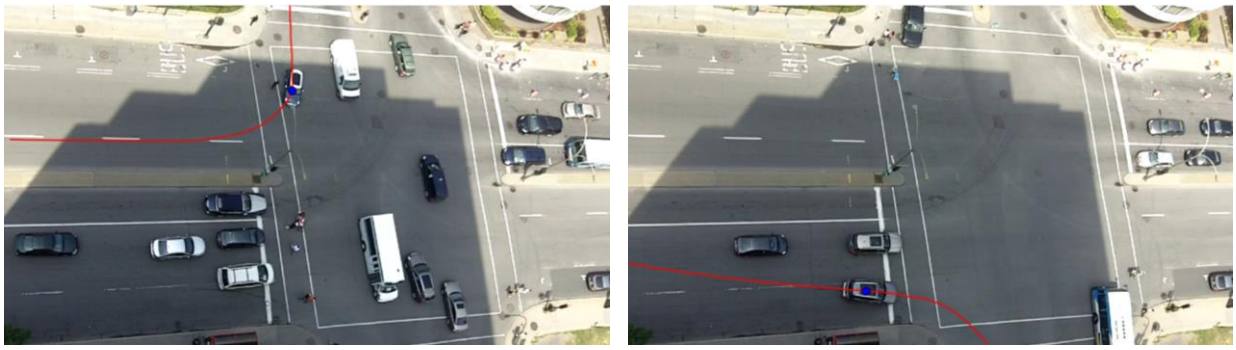
a) Misclassification as vehicle (cyclist)

b) Misclassification as vehicle (pedestrian jogging)

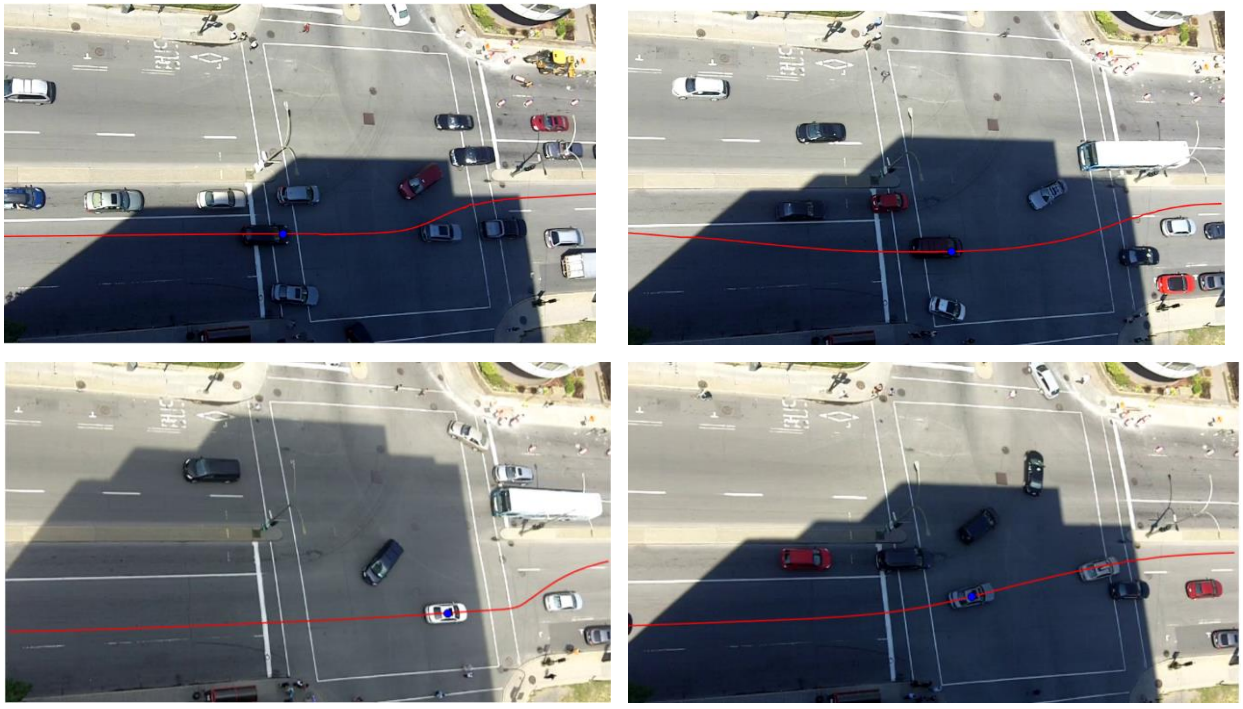


b) Abnormal Left Turn

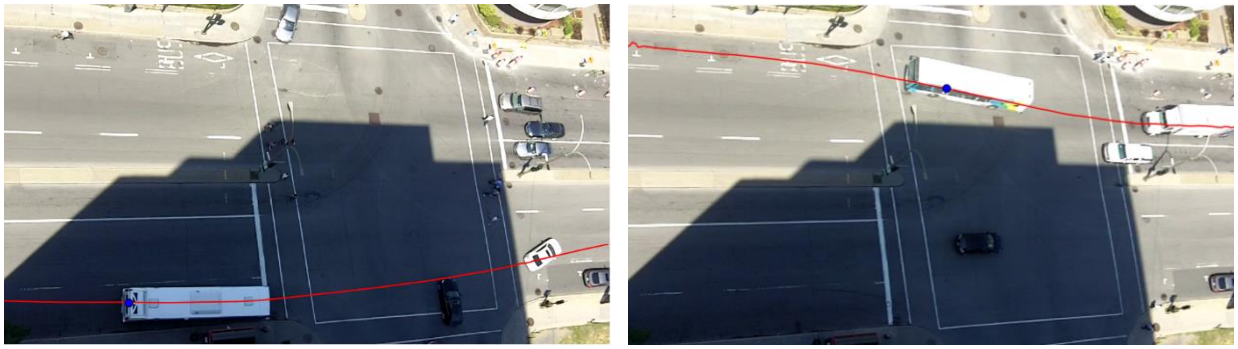
Figure 4.16: Samples of detected spatial anomalies at the Guy intersection



c) Abnormal Right Turn

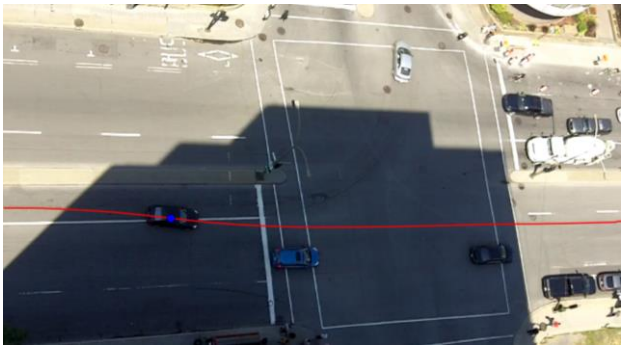


d) Abnormal lane Changes

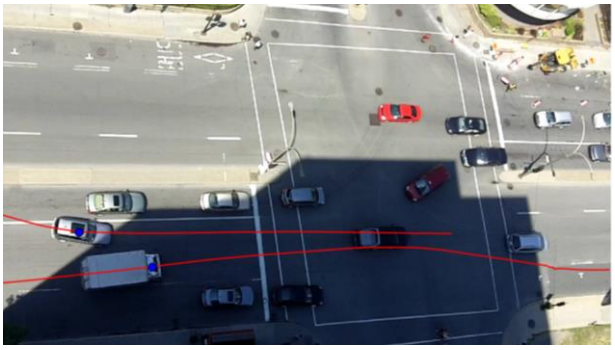


e) Normal movements that rarely happens (bus lane change to its stop station or to its exclusive lane)

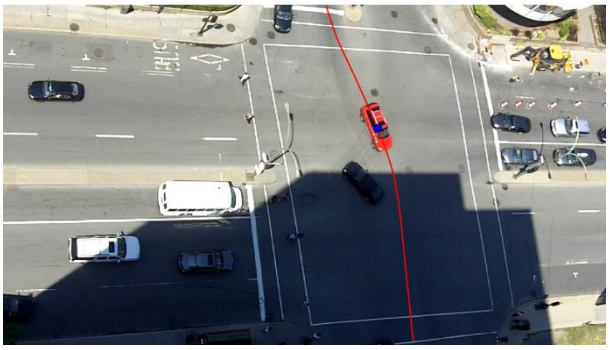
Figure 4.16: Samples of detected spatial anomalies at the Guy intersection (cont'd)



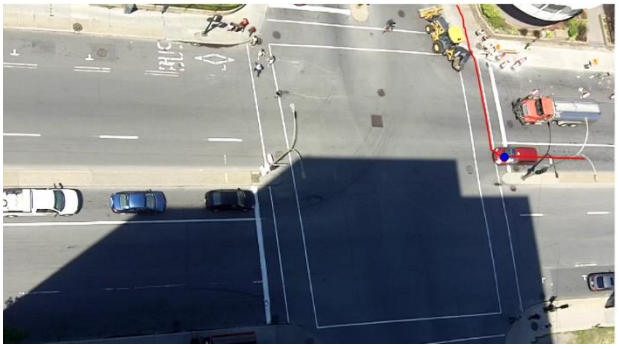
f) Lane change over solid lane markings with no surrounding vehicle



g) Side-swipe interactions (one vehicle changes lane over a solid lane marking while another vehicle adapts its path at the same time)

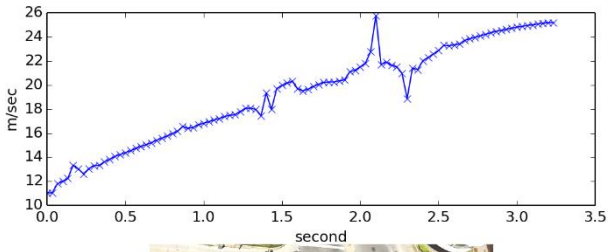


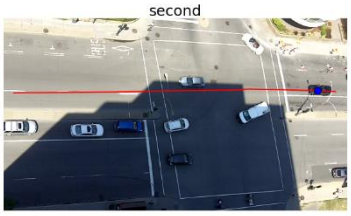
h) Abnormal Overtaking



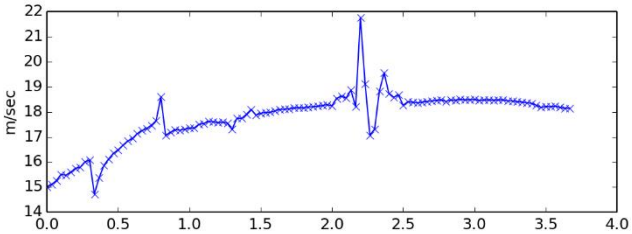
i) Connection error

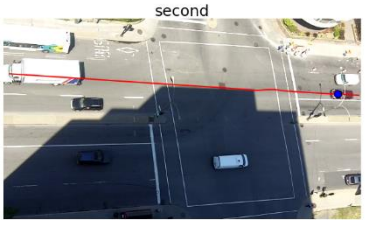
Figure 4.16: Samples of detected spatial anomalies at the Guy intersection (cont'd)





a) Excessive speed (maximum 85km/hr)



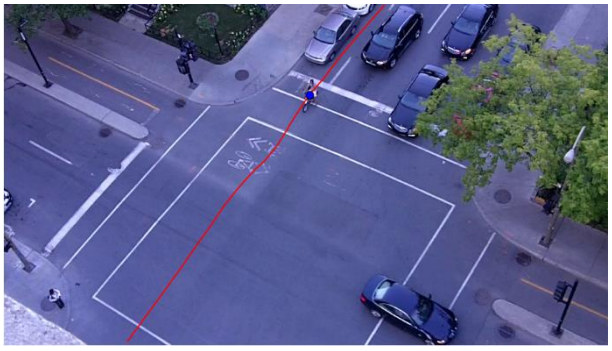


b) Excessive speed (maximum 65km/hr)

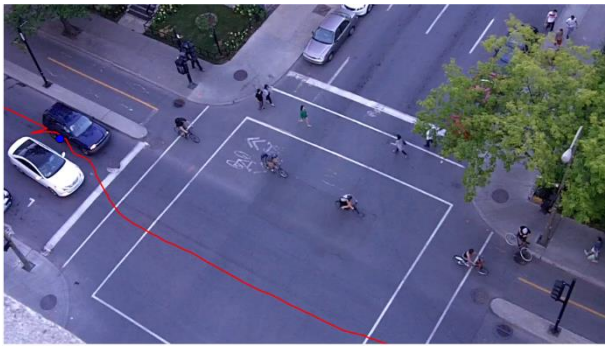
Figure 4.17: Samples of detected excessive speed profile anomalies at the Guy intersection

Figure 4.18 shows samples of detected anomalies in St Marc intersection. Different sources of anomalies are detected such as misclassifications, grouping errors and connection errors. In this

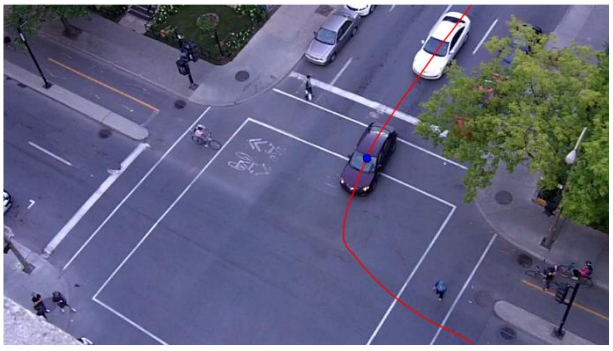
dataset, significant anomalies due to connection errors can be found. The same finding is noted in the cleaning algorithm based on POIs. A possible justification of these errors is the camera FOV where the camera is focused to only cover the central area of the intersection. In the case of connected trajectories, they consist of two segments: the first segment represents trajectory where only part of the vehicle appeared, while in the second segment the entire vehicle appeared. Thus, the vehicle centroid in the first segment is falsely located, and conversely, it is correctly located in the second segment. When connecting these two segments, there is a shift between centroids in connection locations (see Figure 4.18 f)).



a) Misclassification (Motorcyclist)

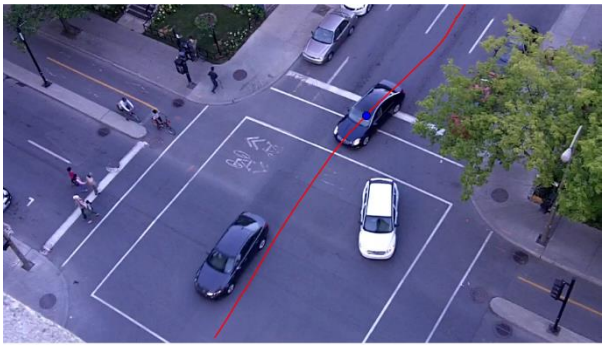


b) Over-grouping error

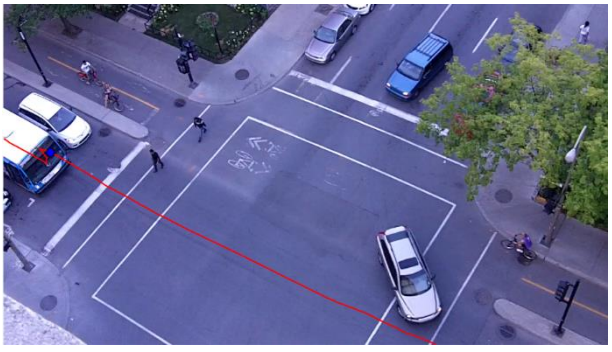


c) Abnormal left turn (not a comfortable or smooth turn: the driver turns at almost 90º)

d) Abnormal Right Turn



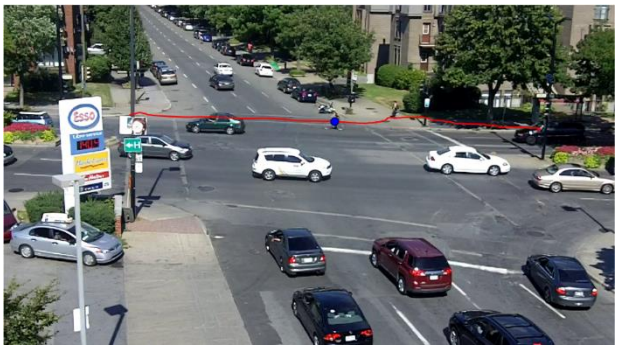
e) Lane Change and overtaking



f) Connection error

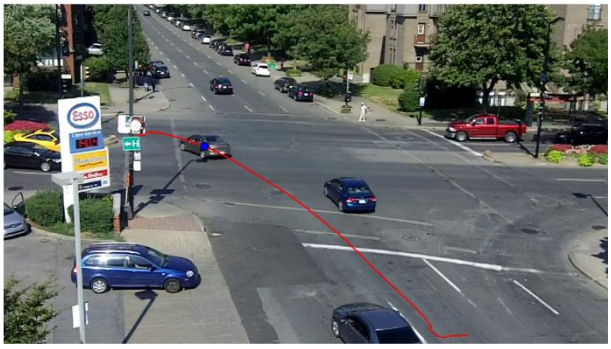
Figure 4.18: Samples of detected spatial anomalies at the St Marc intersection

For the Atwater intersection, there is a large number of anomalies. Most of them are trajectories having an over-grouping problems and some are vehicles projected in the sidewalk. This occurs due to the low angle FOV. By reviewing the detected anomalies manually, we can filter anomalies as summarized in Figure 4.19.



a) Misclassification (cyclist)

b) Over-grouping error



c) Abnormal Left turn

d) Right turn with overtaking a pedestrian

Figure 4.19: Samples of detected spatial anomalies at the Atwater intersection

4.5 Summary

In this chapter, case studies for the application of the multi-level MP framework are presented. The main input of our framework is the trajectory dataset extracted from video using an open source tool “Traffic Intelligence (TI)”. The proposed smoothing algorithm performs well for smoothing vehicle trajectories unless trajectories have over-grouping errors. A comprehensive experimental analysis of three varied scenes demonstrates the ability of the proposed approach to reduce the computation cost in time and required storage space for MP learning, to connect fragmented trajectories effectively, to clean the trajectory dataset from tracking outliers, and to use actual trajectories as prototypes without any pre- or post-processing. These prototypes are used for anomaly detection and can be used to predict a realistic future behaviour as will be explained in the following chapters.

CHAPTER 5 RESEARCH METHODOLOGY (PHASE TWO) SUUROGATE SAFETY ANALYSIS: AN APPLICATION OF MP LEARNING FRAMEWORK AND COMPUTER VISION TECHNINQUE

5.1 Overview

This chapter details the second phase of our research which concerns the applications of computer vision and scene interpretation for safety analysis. This is developed as a generic and consistent safety analysis framework. The definition of safety relevant traffic events relies on the existence of a collision course. This requires specifying a method to predict road users' motions in order to evaluate if they are on a collision course, and to compute several surrogate safety indicators such as TTC. Most analyses rely on the rarely specified or justified method of extrapolation at constant velocity, while several possible paths may in general lead road users to collide.

The main objective of this chapter is to present different methods for motion prediction to predict potential collision points and compute several safety indicators. The general approach follows four main steps and is shown in Figure 5.1:

1. The trajectory of each road user is extracted from video recordings using TI tool (refer to Chapter 3)
2. For each **interaction**, different motion prediction methods are used to predict the future road users' trajectories.
3. At each instant, two predicted trajectories for two road users may have three outcomes: no intersection or an intersection that can be either a crossing zone or a collision point. A **crossing zone (CZ)** is a location in which two trajectories intersect each other. CZ may be between two predicted trajectories, or a predicted trajectory and a road user past trajectory (actual trajectory). A **collision point (CP)** is a crossing zone that the road users are predicted to reach at the same time.
4. The following safety indicators are stored: TTC for each collision point, and predicted PET (pPET) for each crossing zone.

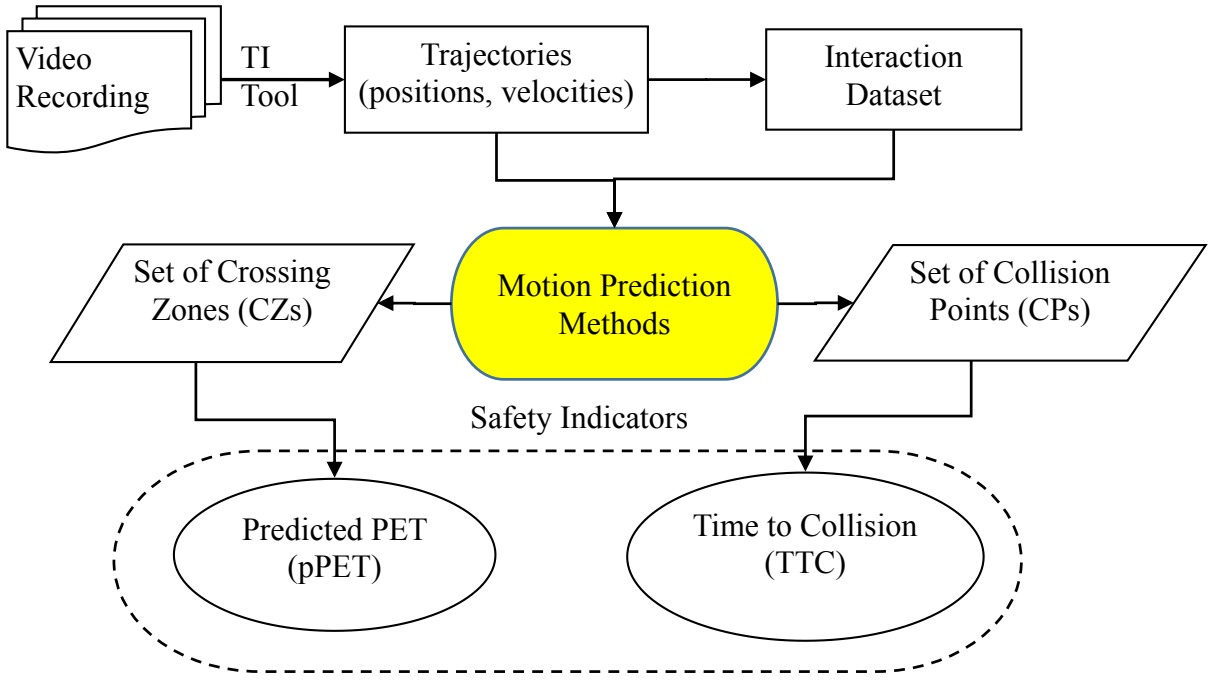


Figure 5.1: Methodology overview

5.2 Motion Prediction Methods

We propose two families of motion prediction methods:

1) *Kinematic methods* are the simplest methods: they are described as context-free prediction methods,

2) *Motion pattern matching methods* are more advanced: the prediction is based on prior knowledge of typical motions in a studied site.

Each family has sub-models with different prediction hypotheses. In this section, the implementation and the performance of each sub-model are investigated in a case study of Left Turn and Opposite Direction (LTOD) interactions as shown in Figure 5.2. Because motion prediction is done at each instant discretely over time in the future, a time horizon for prediction must be chosen and is 150 frames (5 s at about 30 frames per second) in this work.

5.2.1 Kinematic Methods

These methods are more or less complex and depend on the representation of the vehicle kinematics and how uncertainties are handled. However, they do not take into account the geometry of the

road or the presence of other road users (traffic) and can therefore be described as context-free prediction methods. As presented in our previous paper (Mohamed & Saunier, 2013a), the road users' predicted trajectories are determined by their current state and the chosen control input. Similarly to (Broadhurst, Baker, & Kanade, 2005), the current state at t_0 is represented by the state $S(t_0)=(x(t_0),y(t_0),v(t_0),\theta(t_0))$ where $(x(t_0),y(t_0))$ represents the position (if an object is simply represented by its centroid) and $(v(t_0),\theta(t_0))$ are the norm and angle of the velocity vector $(v_x(t_0), v_y(t_0))$. The control input $I(t_0)$ reflects the action undertaken by the road user at t_0, such as acceleration, steering, etc. The $I(t_0)$ vector can be written as $(a(t_0),\Delta\theta(t_0))$ with $a(t_0)$ the acceleration (positive or negative for braking) and $\Delta\theta(t_0)$ the change in the road user orientation both chosen by the road user at t_0. $\Delta\theta(t)$ can be computed as a function of the steering angle $\varphi(t)$, the wheelbase L and the speed $v(t)$ in case of a vehicle as follows:

$$\Delta\theta(t) = \frac{v(t)}{L}\sin(\phi(t))$$

The general formula used to iteratively compute the future position at each time step $t \geq t_0$, where t is discretized at regular intervals Δt, is:

$$\begin{bmatrix} x(t+1) \\ y(t+1) \end{bmatrix} = \begin{bmatrix} x(t) \\ y(t) \end{bmatrix} + \begin{bmatrix} v_x(t+1) \\ v_y(t+1) \end{bmatrix}$$

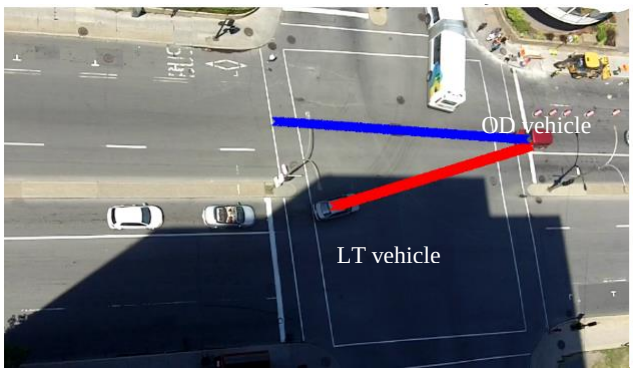
where $\begin{bmatrix} v_x(t+1) \\ v_y(t+1) \end{bmatrix} = \begin{bmatrix} (v(t)+a(t))\cos(\theta(t)+\Delta\theta(t)) \\ (v(t)+a(t))\sin(\theta(t)+\Delta\theta(t)) \end{bmatrix}$

For realistic results, the speed is bounded by 0 and a maximum value v_{max} (i.e.: $v(t+1)$ is the minimum of v_{max} and $v(t)+a(t)$). This model is generic and can represent complex motions, by having varying control inputs $I(t)$ at future time steps $t \geq t_0$. Three kinematic methods are considered to predict possible trajectories and evaluate whether road users are on a collision course or not at t_0:

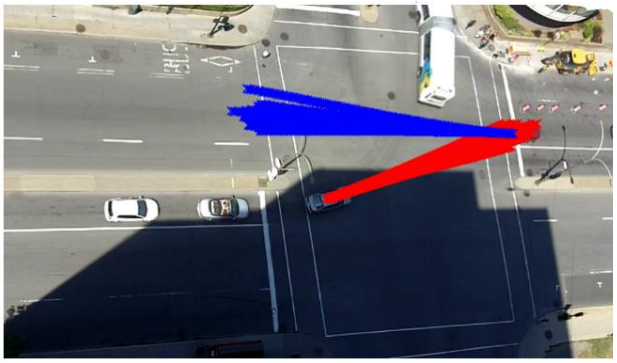
1. **Constant Velocity (CV)**: in case of CV, there is only one predicted trajectory with I(t)=(0,0) for all $t \geq t_0$;
2. **Normal Adaptation (NA)**: in reality, road users make consciously or not small speed and steering adaptation, even when following a straight traffic lane. Such a trajectory can be predicted by drawing the acceleration and orientation change a(t) and Δθ(t) randomly and independently at each step $t \geq t_0$;

3. **Set of Initial Positions (PS)**: if the road user position is represented by a set of positions instead of only its centroid, these can be used as initial position for predicted trajectories. For simplicity and faster computation, prediction is done at constant velocity for each initial position.

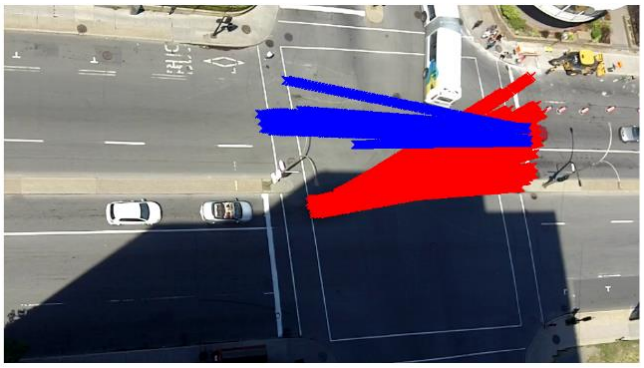
Assumptions are made for reasonable distributions of control input for normal adaptation. Information on this topic is limited in the literature. In (Hydén C. , 1996) (cited in (Archer, 2005)), thresholds on the deceleration-to-safety safety indicator are proposed to measure the conflict severity. Since braking in the range [0,-1m/s^2] and [-1m/s^2,-2m/s^2] was considered to require respectively only "normal adaptation" and a "reaction", the range of [-2 m/s^2, 2 m/s^2] was chosen for normal acceleration in this thesis. The range [-0.2 rad/s, 0.2 rad/s] was chosen for $\Delta\theta(t)$ after some trial and error. The triangular distribution was selected to represent lower probabilities of choosing the most extreme values, with 0 for the mode. These choices could easily be adjusted if better information becomes available. For each road user, N_1 predicted trajectories are generated for the normal adaptation method. Figure 5.2 shows the predicted trajectories in an interaction example using the three kinematic methods. It is noticed that the left turning (LT) vehicle is moving in a direction that could collide with an opposite direction (OD) vehicle moving straight according to these methods.



(a) Constant Velocity Method



(b) Normal Adaptation Method (N_1 = 10)



(c) Set of Initial Position Method

Figure 5.2: Examples of predicted trajectories using kinematic methods (the blue and red crosses or lines in each image).

5.2.2 **Motion Pattern Matching (MPM) Methods**

In these methods, road user motion is matched to learnt prototype trajectories representing typical motion patterns at a studied site. The predicted trajectories follow the matched prototypes based on the current road user speed. MPM is an application of the multi-level motion pattern learning framework described in Chapter 3 and investigated for different sites in Chapter 4. MPM models have the advantage of taking into consideration site geometry and traffic environment compared to the previous context-free motion prediction methods (for example, most road users will not continue straight into a curb or a wall).

Given a set of prototypes and their matching counts (in other word "their cluster size"), future motion can be predicted with their associated probability. Let T be the trajectory of a road user entering the scene and T_{t_0} the partial trajectory up to the current frame (instant, t_0). The partial trajectory is compared to all the prototypes with the same entry zone and future positions are

predicted based on them. Only prototypes ϕ_i of which similarity exceeds a defined threshold are considered as a possible future trajectory. The similarity is computed based on the LCSS measure with a defined δ (selected as 60 frames). The probability of following each prototype is estimated using the Bayes rules:

$$P(\phi_i|T_{t_0}) = \frac{P(T_{t_0}|\phi_i)\,P(\phi_i)}{\sum_{n=1}^{N} P(T_{t_0}|\phi_n)\,P(\phi_n)}\,, \qquad i = 1, 2, \dots N$$

Where $P(\phi_i)$ is estimated by the ratio of the cluster size corresponding to ϕ_i to the sum of the cluster sizes of all the matched prototypes. $P(T_{t_0}|\phi_i)$ is measured by the SLCSS (normalized similarity measure). This factor is used to weight the estimated probability based on the similarity between T_{t_0} and ϕ_i. An empirical example is shown in Table 5.1.

Table 5.1: An example of a probability estimation

	Cluster size	$P(\phi)$	$P(T_{t_0}\|\phi)$	$P(\phi\|T_{t_0})$
ϕ_i	70	0.7	0.9	0.8
ϕ_j	30	0.3	0.5	0.2

In order to obtain a set of predicted trajectories covering the possible driver behaviour based on MPM, two main concepts are considered:

1. **Using the destination**: a road user usually has a destination in mind, at least at the typical scale of the studied zones, and one should not predict that the road user may leave the scene through another exit than the one he actually took. Using the entry/exit zones (detected using the POIs), the destination of each road user is known if the trajectory is complete, and the partial trajectories will be matched with the prototypes that have the same destination. This assumption shows the importance of POIs.
2. **How to use the speed profiles**: the current speed of each road user may be used in two ways. The first way is to assume the road user will move with constant current speed. It means that the road user will follow the matched spatial prototypes with constant speed. The other way is to consider the speed profiles in each MP. That means the future positions are predicted based on the learnt temporal prototypes instead of the spatial prototypes. The matched prototypes are translated to the centroid of road user and resampled using the current speed of the road user as performed in (Saunier, Sayed, & Lim, 2007).

The procedure of motion prediction follows the presented the flow chart in Figure 5.3. It is divided into two main steps: step (1) is responsible to filter which set of spatial prototypes will be compared with the partial trajectory. It relies on the concept of using destination or not to select the set of prototypes. Step (2) is used to find the matched spatial prototypes whose similarity to the partial trajectory exceeds a threshold. In the prediction step, the future trajectories will follow the positions of the spatial prototypes with constant current speed when using the constant speed hypothesis or the positions and resampled speeds of temporal prototypes belonging to the matched spatial prototypes when using the speed profiles hypothesis. Accordingly, the former hypothesis leads to the same travelled distance for all predicted trajectories, while the latter hypothesis generates predicted trajectories with varied travelled distances over time.

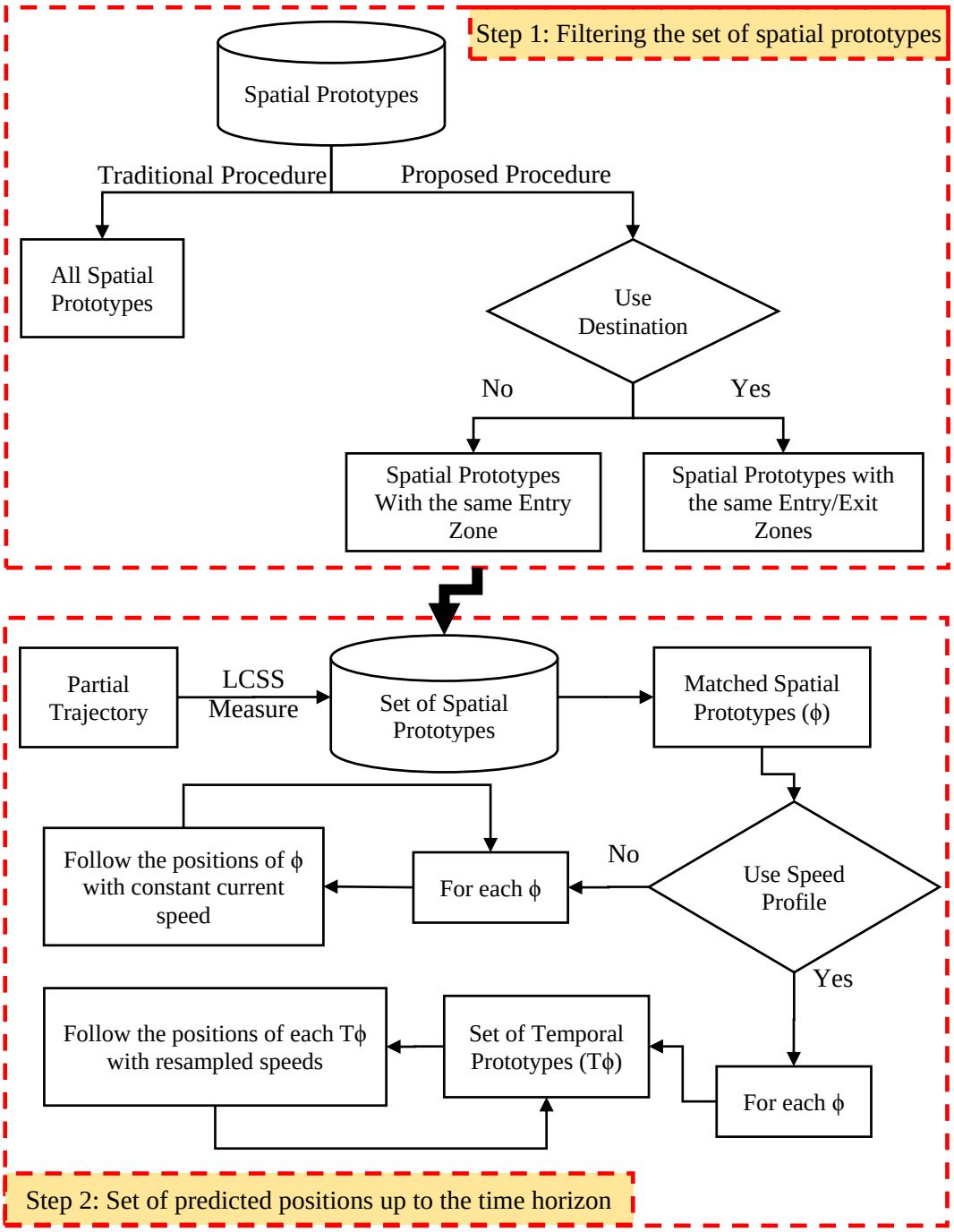


Figure 5.3: A flow chart of MPM prediction procedure

Figure 5.4 shows examples of motion prediction for the LTOD interactions based on different methods using MPM. The methods rely on different combinations of hypotheses of using destination and/or using speed profiles. It is clear that the main difference of using the speed profile or not is the travelled distances of the predicted trajectories. In addition, it is noticed that some of predicted trajectories will follow the through movement or turn right for the LT and OD vehicles

respectively when not using of destination hypothesis. The most important finding is that the predicted trajectories follow a realistic motion, in particular, the LT vehicle makes a turn and does not move in the wrong direction as noticed with the kinematic models. Examples of predicted speed profile for both LT and OD vehicles are shown in Figure 5.5. These speed profiles belong to the case of predicted trajectories using the speed profile and the destination hypotheses for the same LTOD interaction. As shown in the figure (especially for left turn), some profiles have constant speed at the end. This occurs when the length of speed profiles in a matched temporal prototype is less than the time horizon, this length is measured by the number of instants from the current instant till the last instant of the prototype (when the scene limit is reached). In this case, we solve the limited availability of speed data by using the last instant speed the constant speed hypothesis till the end of the time horizon. Some of the predicted positions extend out of the scene limit, but only potential collisions observed within the scene limits are considered in the safety analysis.

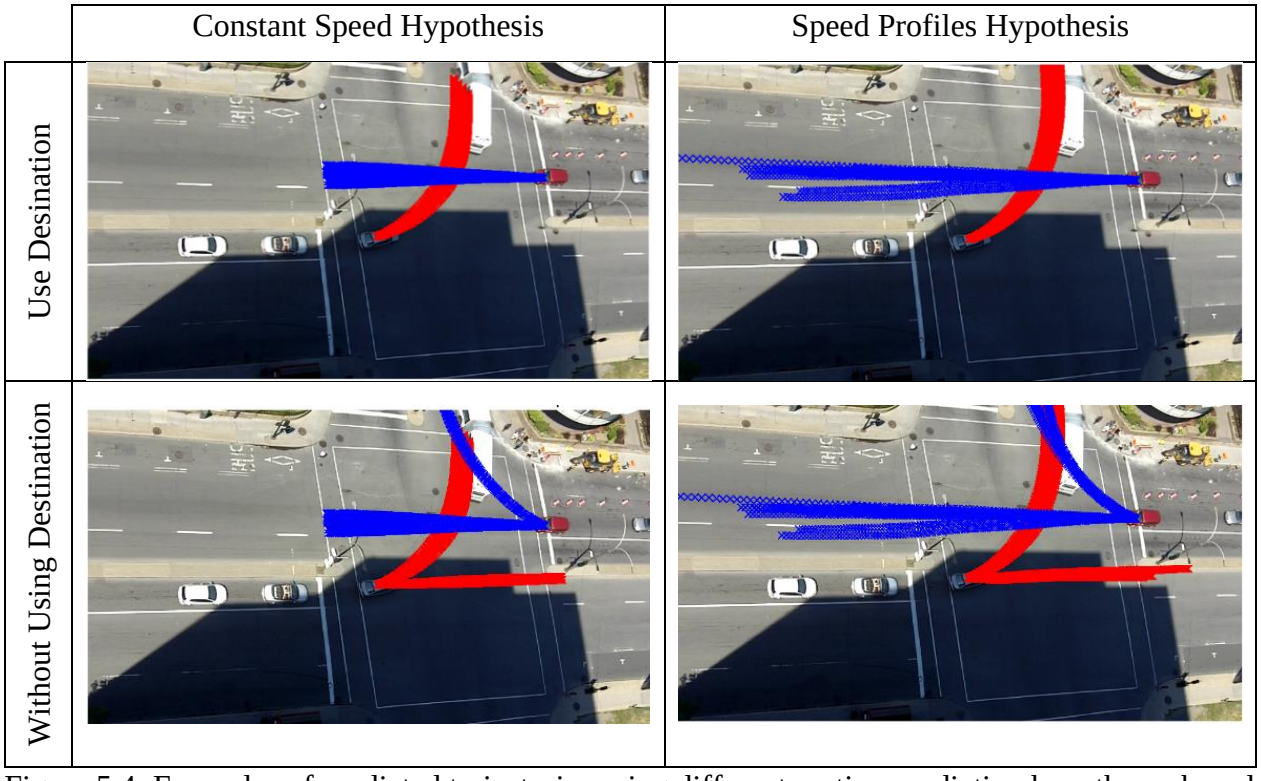
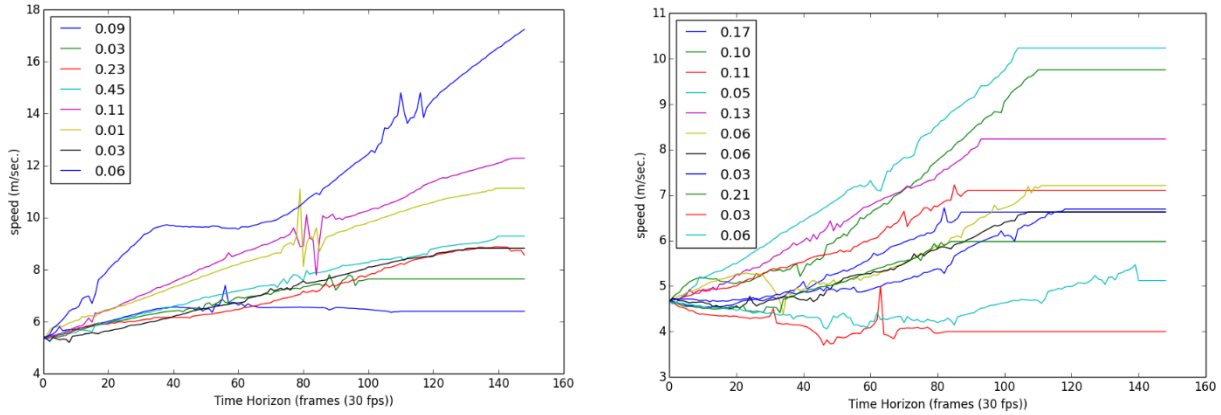


Figure 5.4: Examples of predicted trajectories using different motion prediction hypotheses based on MPM (the blue and red crosses or lines).



(a) Through Movement (b) Left Turn Movement

Figure 5.5: Examples of predicted speed profiles for the through and left turn movements for the same stated interaction and their associated probability in the legend of each figure

5.2.3 Comparison between Different Motion Prediction Methods

The aforementioned prediction methods provide three kinematic models and four MPM based prediction methods. A comparison between the different methods is conducted based on the required information and running time required to predict the trajectories of two interacting road users at any instant. The required information is defined as any assumption or data needed other than the current position and velocity for each road user in an interaction. Moreover, the performance of each type of method with respect to realistic motion prediction is discussed in the previous sections. The comparison is summarized in Table 5.2. It should be noted that the running time is computed only for the prediction procedure while the running time of learning the prototypes in the four MPM methods is neglected in the comparison because it is a one-time procedure. Moreover, the running time is a function of the number of predicted trajectories N_1 for the NA method, the number of features N_2 for the PS method, and the number of prototypes for the MPM methods. The prototypes are divided into two datasets based on the two main steps shown previously in Figure 5.3. The first dataset is the set of spatial prototypes N_3 based on the entrance zone (and exit zone if using destination) and, the second dataset contains the N_4 matched spatial prototypes or temporal prototypes used in the prediction step (refer to step2 in Figure 5.3). That is why N_3 is larger without destination, and N_4 is larger when using temporal prototypes since for each spatial prototype, there is a set of temporal prototypes used for prediction. In an interaction, two datasets for each interacting road user influence the running time.

Regarding the four MPM methods, a road user enters the scene in a known entry zone and its partial trajectory (at each instant) needs to be matched only to the MP prototypes that share the same entry zone without the need to compute the similarity to all the MP prototypes in the scene as performed in past approaches. This simple procedure is efficient to speed up motion prediction. Besides, if destination is known and used, the partial trajectory is matched only to the prototypes with the same entry and exit zones. A motion prediction using traditional MPM procedure (computing the similarity to all prototypes without considering the entry/exit constraint) is investigated to test the effectiveness of the proposed MPM method in speeding up the computation of motion prediction while providing the same results (refer to Table 5.2). The running time is estimated by taking the average of 10 times runs of generating predicted trajectory for the two interacted vehicles.

Table 5.2: Comparison between different motion prediction methods in term of required information and running time for two road users in a given interaction at a given instant.

Motion Prediction Method	**Required Information**	**Average running time in seconds (Standard Deviation)**
KM-1 Constant Velocity	None	0.026 (0.003)
KM-2 Normal Adaptation	Steering & acceleration distributions Number of predicted trajectories	2.76 (0.18) N_1=100 per road user
KM-3 Set of Initial Positions	None	0.92 N_2=33 per LT vehicle N_2=13 per OD vehicle
MPM-1 Use destination Constant speed	Entry/Exit zones Spatial prototypes	11.83 (0.07) N_3=11 & N_4=11 per LT vehicle N_3=24 & N_4=8 per OD vehicle
MPM-2 Use destination Use speed profiles	Entry/Exit zones Spatial and temporal prototypes	25.21(1.94) N_3=11 & N_4=13 per LT vehicle N_3=24 & N_4=20 per OD vehicle
MPM-3 Without destination Constant Speed	Entry/Exit zones Spatial prototypes	34.98(1.97) N_3=21 & N_4=14 per LT vehicle N_3=39 & N_4=10 per OD vehicle
MPM-4 Without destination Use speed profiles	Entry/Exit zones Spatial and temporal prototypes	49.11(0.73) N_3=21 & N_4=19 per LT vehicle N_3=39 & N_4=24 per OD vehicle
MPM-5 (traditional procedure) Without destination Constant speed	Spatial prototypes	83.23(7.09) N_3=133 & N_4=14 per LT vehicle N_3=133 & N_4=10 per OD vehicle
MPM-6 (traditional procedure) Without destination Use speed profiles	Spatial and temporal prototypes	96.91(5.78) N_3=133 & N_4=19 per LT vehicle N_3=133 & N_4=24 per OD vehicle

As shown in Table 5.2, the running time required to predict trajectories for the kinematic methods is significantly lower than the running time required for MPM methods. CV proves to be the fastest method and has a significantly smaller running time than other methods. For MPM methods, our proposed method is able to accelerate the running time significantly. To compare between performance of our method and traditional procedures; MPM-3 is compared with MPM-5 and MPM-4 is compared with MPM-6. It is noteworthy that the gain in running time is between 50 to 60 % compared to traditional procedures. When using the POIs and the destination as constraint; the running time is accelerated and the gain reaches around 74 to 86 %.

5.3 Automated Safety Indicators Calculations

The rich microscopic positional data extracted by video data and computer vision help to understand the traffic behaviour using a variety of indicators. In this thesis, the indicators are grouped into two main categories: simple indicators and surrogate safety indicators. The following section describes each category in detail.

5.3.1 Simple indicators

Simple indicators are computed at each instant within the interaction interval. They are classified into two sub-categories as follows:

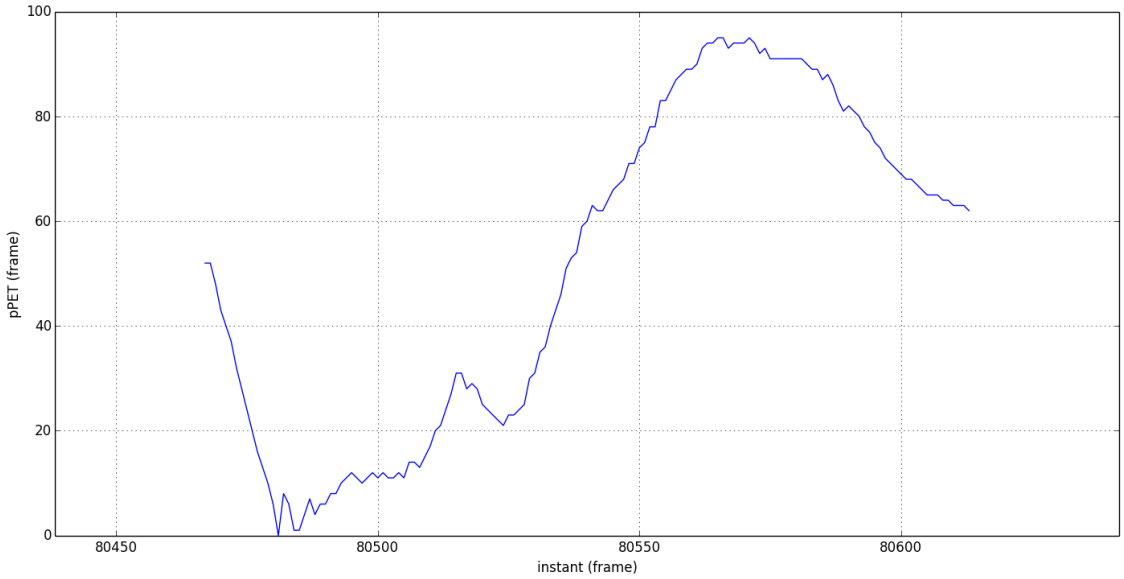
- ***Individual road user indicators***: This type includes mainly kinematic profiles (i.e. speed and acceleration profile) which describe each road user's individual behaviour during an interaction. These indicators are derived from positional data as the first and second time derivative of the positions. The speed is an important factor in road safety as it can affect the severity of a collision. The acceleration can represent the magnitude of evasive actions taken by road users in an interaction. In most studies, the acceleration rate when negative, i.e. the deceleration rate, corresponds to an evasive action taken by one or two interacting road users. The noise generated in the tracking process lead to higher level of noise in derived acceleration profiles. Therefore, the second-order degree Savitzky-Golay filter (Savitzky & Golay, 1964) is used to smooth and filter the acceleration profile. This filter is applicable to time-series data with regular time step.

- ***Interaction indicators***: As shown in (Saunier, Sayed, & Ismail, 2010), an interaction can be well described by several symmetrical indicators independently of the road users' absolute positions. Such indicators make it easier to compare interactions occurring in different areas of the traffic scene under study. For this thesis, one spatial indicator and one speed indicator are computed: the distance (dist) between the road users, and the speed differential (SD), i.e. the norm of the difference of the velocities. The distance is measured using the Euclidean distance between the road users' centroids. More precisely, it can be calculated as the minimum distance (MinDist) of all distances between each road user's features. By construction, "MinDist" is less than "Dist" and takes better into account the road users' volumes.

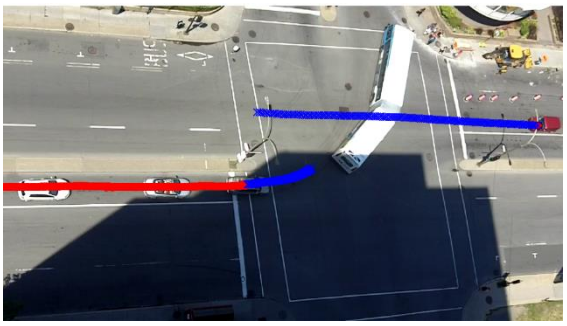
5.3.2 Surrogate Safety indicators

Several quantitative safety indicators were developed to measure the proximity of an interaction to a potential collision. Among them, two sub-categories of surrogate safety indicators used in the literature will be investigated:

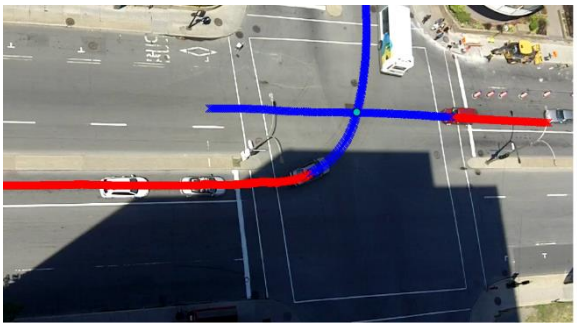
- ***Direct (single) measurable indicators***: PET is a common example of an indicator that is measured directly using the actual trajectories: it is defined as the time difference between the instants that the two road users leave and enter a crossing zone. One of the shortcomings of PET is that the estimation of proximity to a collision is difficult because of the lack of speed and distance dimension with the indicator value.
- ***Estimated (continuous) measurable indicators***: These indicators, such as TTC and pPET, rely on the motion prediction methods at each instant during an interaction. pPET is defined as the difference between the times at which the road users would reach the predicted CZ. The calculation of pPET is associated with the calculation of CZ, and therefore can be computed in two cases as shown in Figure 5.6: 1) between two predicted trajectories or 2) between a predicted trajectory and an actual trajectory.



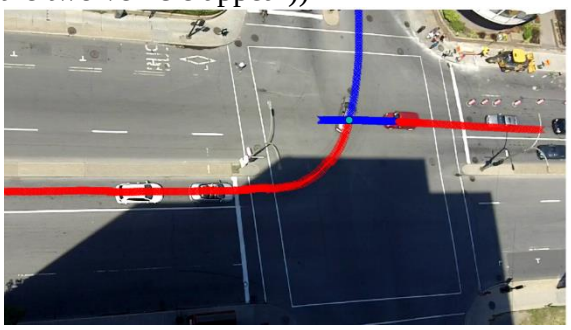
(a) The evolution of pPET over time



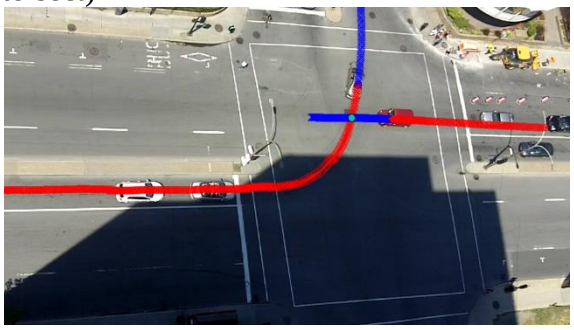
(b) Instant 80431 (interaction starts when the two vehicle appear))



(c) Instant 80500 (two predicted trajectories intersect)

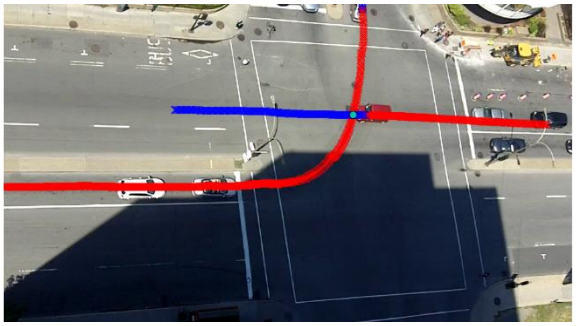


(d) Instant 80562 (the first vehicle reaches the predicted CZ)

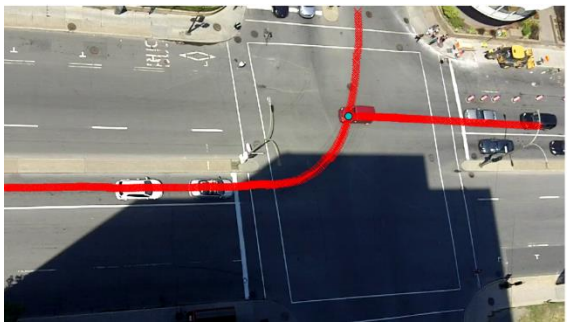


(e) Instant 50580 (a predicted trajectory of the second vehicle intersects the actual past trajectory of first vehicle)

Figure 5.6: An example of the pPET evolution during an interaction interval (red crosses refer to past trajectories, blue ones refer to predicted trajectories, and the cyan point is the predicted CZ)



(f) Instant 80613 (interaction ends where first vehicle leaves the scene)



(g) Instant 80622 (actual trajectories intersect at the CZ and PET is computed)

Figure 5.6: An example of the pPET evolution during an interaction interval (red crosses refer to past trajectories, blue ones refer to predicted trajectories, and the cyan point is the predicted CZ) (cont'd)

To compute these indicators, it is important to select the road users' appropriate representation and the choice of motion prediction methods. The methodology of computing surrogate safety indicator based on these two factors is discussed in details in the following sections.

5.3.2.1 Geometric Shape Representation

Four different automated measurement methods are proposed that rely on each road user's assumed geometric shapes. These methods are shown in Figure 5.7. As readers could expect, the complexity of the assumed shape has a significant influence on the required running time. The different representations will be tested and demonstrated on the computation of PET.

[1,2] **Centroid Based Methods:** these methods represent the road users by their centroids. There are in fact two methods with different implementations. Although the assumption of road user as a point over-simplifies the representation of a road user volume, it enables to compute some indicators with expensive computation cost. In addition, this method could be used as pre-screening procedure before using more detailed geometric representations. Another way to look at the centroid representation is through the use of thresholds to detect collision points or crossing zones, in which case one can consider road users to be represented by circles. Defining the diameter of any circle (or threshold) is a very challenging task. Considering the diameter as equal to the road user's width or length underestimates or overestimates respectively the road user's volume.

Identifying crossing zones implies testing the intersection of all segments constituting a trajectory: the number of computations can therefore reach the product of the lengths of the two trajectories.

Alternatively, the PET may be computed based on measuring distances between all positions in any two trajectories as follows:

i. Construct a distance matrix between all positions in both trajectories. The distance matrix is calculated using a built-in function in SciPy library (an open source Python library). The function is called cdist (see http://docs.scipy.org)

    a. In the first method called CBM, the algorithm determines the two instants (one instant per trajectory) corresponding to the shortest distance. If the distance is small enough, the difference between those two instants is the PET value.

    b. In the second method called CirBM, all PETs corresponding to distances inferior to the threshold (circle's diameter) are computed and then take the minimum PET.

The difference between CBM and CirBM is in their implementation: only one PET is computed in the case of CBM while many PETs may be computed in the case of CirBM. As a result, the running time is significantly smaller for CBM than for CirBM.

[3] **Features-Based Convex Hull method (FBCH)**: This method considers the road user as a polygon computed as a convex hull which represents the road user's outline. The estimated polygon is computed as the convex hull of the detected features obtained during tracking. However, a convex hull may not represent all edges of a road user in case of missing features (e.g. see the detected features for dark vehicle in Figure 5.7 a)). It should be clarified that this shortcoming will also affect the estimation of the centroid. In this case, all the PETs at which the two interacting road users represented by convex hull polygons overlap each other are computed and then the minimum PET is the final measurement. The implementation of FBCH method relies on the built in "ConvexHull" function in the scipy library and the "Polygon" function in the shapely library. The former is used to create the convex hull, the latter to check if two road users overlap each other.

[4] **Feature-based Minimum Area Bounding box method (FMAB):** This method represents a road user (for vehicle type) as a 2D rectangle, known as a Bounding Box (BB). BB is traditionally computed using the minimum and maximum coordinates to identify the width and the length of the rectangle without considering its rotation, thus, it is usually bigger than the actual size. A possible estimation of the BB is to find the rectangle drawn with minimum area. This is performed using the function "minAreaRect" in the opencv python library (see http://docs.opencv.org/). The FMAB

algorithm works at each frame (instant) as follows (note that step ii is performed in the minAreaRect function):

i. Compute the convex hull of the features of each road user.

ii. For each edge of the convex hull,

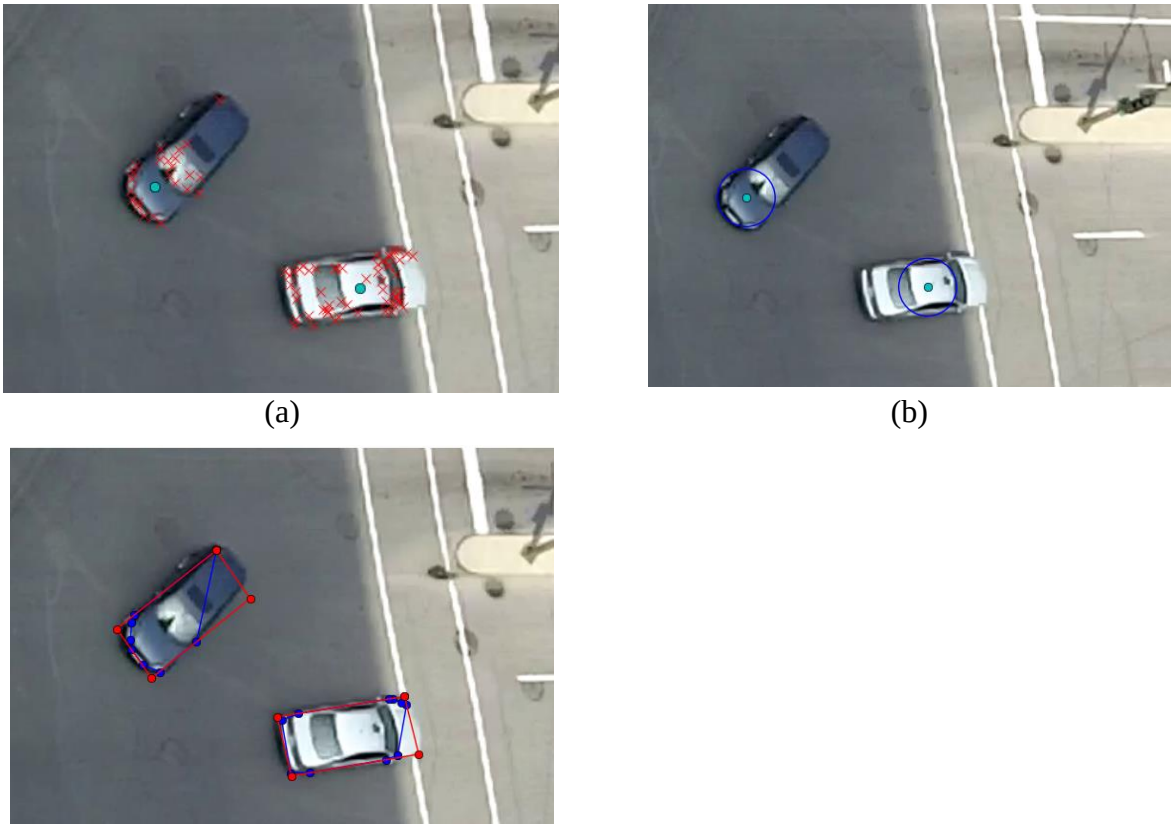
   a. Compute the orientation of this edge (θ)

   b. Rotate the convex hull coordinates by θ.

   c. Compute a BB rectangle using min/max of feature coordinates.

   d. Compute the rectangle area and store the corresponding rotation angle.

iii. All PETs at which the road users' BBs overlap each other are computed and the minimum PET is the final measurement.



(a) (b)

(c)

Figure 5.7: Sample of a geometric representation (a) detected features (red crosses) and computed centroid (cyan circle), (b) the centroid based methods: CBM (cyan point) and CirBM (blue circle) representations, and (c) the FBCH (blue polygon) and the FMAB (the red rotated rectangle) representations.

5.3.2.2 Choice of Motion Prediction Method

Among the proposed motion prediction methods, four methods (two based on kinematic methods and two based on MPMs) are selected to analyze the safety of the interactions in the next chapter. These methods are CV, NA, MPM-1 (using destination and constant speed, will be referred to as MPCS) and MPM-2 (using destination and speed profiles, will be referred to as MPSP). Several other prediction methods could be used, but those four methods provide a variety of predictions that can serve as a basis for investigation and comparison. Finding the collision points and crossing zones at each t_0 consists in predicting the trajectories for each pair of interacting road users over a fixed time horizon (5 s is used). A collision point is identified using CirBM method if the distance between the interacting road users' predicted positions is below a threshold (1.8 m is used in the thesis as this represents the typical width of a car). The future time at which this condition is met is the TTC. If there is no collision point, the algorithm searches for an intersection between the two predicted trajectories or predicted and actual trajectories: if there is an intersection or "crossing zone", the pPET is computed. One can see that TTC and pPET complement each other. Through the course of an interaction, TTC and pPET may be calculated at the same instant, although not in the case of the CV method which has only one predicted trajectory for each road user.

An important topic is the computational cost of generating the predicted trajectories and computing the collision points and crossing zones. Identifying collision points requires a number of tests up to the time horizon, the selected time horizon is 5 s that is 150 steps with a frame rate of 30 frames per second. Identifying crossing zones is more costly as it implies testing the intersection of all segments between two predicted trajectories: the number of computations can therefore reach the square of the time horizon, which is 150^2=22500 in this study. This number of computations is then multiplied by the product of the number of predicted trajectories for the two road users, e.g. N_1^2 for the NA method, which amounts to up 10000x22500 computations at each instant in this study. This is therefore very expensive as the maximum number of intersection tests (for crossing zones) grows with the product of the square of the time horizon and the square of the number of predicted trajectories. To deal with this shortcoming, pPET is estimated using the CBM method by computing a distance matrix between any two predicted trajectories. This procedure significantly reduces the running time but is still too expensive to be applied in all the experiments. In addition, considering a road user as a point reduces the accuracy of the computed indicators by construction. Further work is needed to speed up the process and find the right trade-off between the

computational cost and a good coverage of the potential movements that may lead the road users to collide. To sum up, road users are presented as circles, as in the CirBM method, in the case of TTC computation. On the other hand, in the case of pPET computation, road users are presented by their centroid point similar to the CBM method.

At each time instant t_0, a set of predicted trajectories for two road users may generate a set of collision points and crossing zones, with their associated TTC and pPET. Similar to (Saunier, Sayed, & Ismail, 2010; Mohamed & Saunier, 2013a), the expected TTC and pPET are their expected value over respectively all collision points and crossing zones as shown in the following equations. Note that indicator is weighted by probabilities for each predicted trajectory, i.e. ($P(g_1(--))$ and $P(g_2(--))$: for the NA method, this is implicitly taken into account by the distribution of the control input, however, for MPM methods, this is equivalent to the previously described probability of following each prototype $P(\phi_i|T_*)$.

$$P\big(Collision\ (CP_n)\big) = P(g_1(CP_n))P(g_2(CP_n))$$

$$P\big(Crossing\ (CZ_n)\big) = P(g_1(CZ_n))P(g_2(CZ_n))$$

$$TTC\ \left(U_i, U_j, t_0\right) = \frac{\sum_{1\leq n<N_{CP}} P\big(Collision\ (CP_n)\big)t_n}{\sum_{1\leq n<N_{CP}} P\big(Collision\ (CP_n)\big)}$$

$$pPET\ \left(U_i, U_j, t_0\right) = \frac{\sum_{1\leq n<N_{CZ}} P\big(Crossing\ (CZ_n)\big)\ |t_{i,n} - t_{j,n}|}{\sum_{1\leq n<N_{CZ}} P\big(Crossing\ (CZ_n)\big)}$$

Where: ($P(g_1(--))$ and $P(g_2(--))$ are the probability of the road users follow a particular motion prediction method, $g_1$and $g_2$are the predicted trajectories, $g_1(CP_n)$ and $g_1(CZ_n)$ are the predicted trajectories of the first road user that are involved in CP_n and CZ_n respectively, U_i and U_j are the interacting two road users, and t_0 the current instant.

5.4 Continuous Indicator Interpretation

5.4.1 Indicator aggregation

In most of the current practice and research, continuous indicators are aggregated into only a single value based on the most severe observation (e.g. minimum or percentile value over all observations

per interaction). The analysis of the aggregated values based on different motion prediction methods is performed based on the two following methods:

- ***Distribution analysis:*** this analysis is mainly used as a visual approach (some statistical tests could be used as well) and may support the safety continuum concept. The safety can be estimated by looking at the indicator level belonging to the peak of the distribution. This can be used to compare different sites or before-after studies. However in this thesis, this method is used to evaluate the influence of motion prediction method on the evaluation of the safety of a specific site.
- ***Interaction frequency based on a threshold:*** this method relies on defining a threshold and counting the number of interactions with the indicator more severe than the threshold. The interaction frequency for each motion prediction methods will be compared to find which method and threshold can have an agreement with collision frequency. It should be clarified that finding and validating a relationship between collisions and interactions is out of this thesis scope as it requires the collection of large video dataset and associated collision records.

5.4.2 Indicator Clustering

Narrowing down the whole interaction process to a single value leads to lose a lot of information: Consequently, the indicator clustering analysis goes beyond using only a single value of the indicator to qualify the whole interaction: the whole time series or profiles of the indicators are analyzed to find similarities between interactions. Data mining techniques are used to group interactions into homogeneous groups with similar indicator profiles. This analysis is performed to find the similarity between interactions with and without collision in the first version (Mohamed & Saunier, 2013b) of this work using LCSS as a similarity measure and spectral clustering algorithm. This version was fast and takes as only input the predetermined number of groups. However, the iterative process by trial and error to find the number of clusters proves to be a challenge, and the resulting clusters were not always easy to interpret. Consequently, the method presented in Saunier & Mohamed (2014) overcomes these challenges by proposing the ALCSS similarity measure and using a custom clustering algorithm. These methods were presented in detail in Chapter 3 for temporal prototype learning. The same methodology will be applied for safety indicators classification.

5.5 Summary

Despite the rise in interest for surrogate safety analysis, little work has been done to understand and test the impact of motion prediction methods, of the geometric representation of road user volume, and of the interpretation of safety indicators. Motion predictions are needed to identify whether two road users are on a collision course and to compute many surrogate safety indicators. The default, unjustified method used in much of the literature is prediction at constant velocity. In this chapter, a generic framework is presented to predict the road users' future positions depending on their current position or past trajectory. Two main methods are introduced and compared: kinematic and MPM methods. Kinematic methods are faster than MPM methods. However, the proposed MPM method reduces the running time up to 86 % compared to past procedures of prototype based motion methods. Moreover, it has the advantage of taking implicitly the context into account, such as the road geometry. Four geometric methods are proposed to represent the road users' volume and to compute the PET automatically. A probabilistic versions of two safety indicators, the expected TTC, and a predicted PET are computed for all predicted trajectories. Finally, two indicators interpretation methods are proposed based on aggregated values and on whole profile classification.

CHAPTER 6 EXPERIMENTAL RESULTS (PHASE TWO)

6.1 Overview

This chapter presents an experimental results of the proposed methodology for surrogate safety analysis. The analysis is focused on three topics: 1) analyze the influence of the road users' geometric shape representation on the accuracy of the estimated safety indicators, 2) investigate different motion prediction methods to predict potential collision points and compute several safety indicators such as TTC and pPET, and 3) investigate several interpretation approaches for safety analysis under the proposed generic framework. The proposed framework is tested mainly on one of the case studies used in Chapter 3. This case study is the Guy intersection in Montreal, Canada, in particular, the LTOD interactions and collisions. In addition, two other case studies are used for specific investigations. The Atwater intersection, used as well in Chapter 3, is studied to evaluate the influence of geometric shape representation in the case of low angle FOV. Moreover, a dataset containing a large real world dataset of traffic conflicts and collisions, called the Kentucky dataset, is used for the safety indicator clustering to find similarities between interactions with and without collision. This will demonstrate the proposed framework and allows for the use of the generic method to guide safety analysts to estimate the safety of sites under study and to recognize the types of interactions that require attention.

6.2 Description of the Case Studies

6.2.1 Guy Intersection

The LTOD interactions and collisions occurring in this intersection are the main case study of the proposed safety analysis. The choice is made for three reasons: first, LTOD collisions are generally the most prevalent conflicts and collisions at any signalized intersection (Najm, Smith, & Smith, 2001; Ragland & Zabyshny, 2003) and specifically in the studied intersection: they resulted in 36 percent of the collisions (15 collisions out of total 42) in the period of 2005-2007 and the intersection was not modified till August 2012 when this data was collected. Secondly, predicting future motions for turning movements is more challenging than for the other movements. Finally, constructing simple bounding boxes for turning movement leads to larger representations of the road users. Consequently, a good performance of the proposed methodology, in particular the

performance of motion prediction in the LTOD interactions, should indicate good performance for other types of interactions.

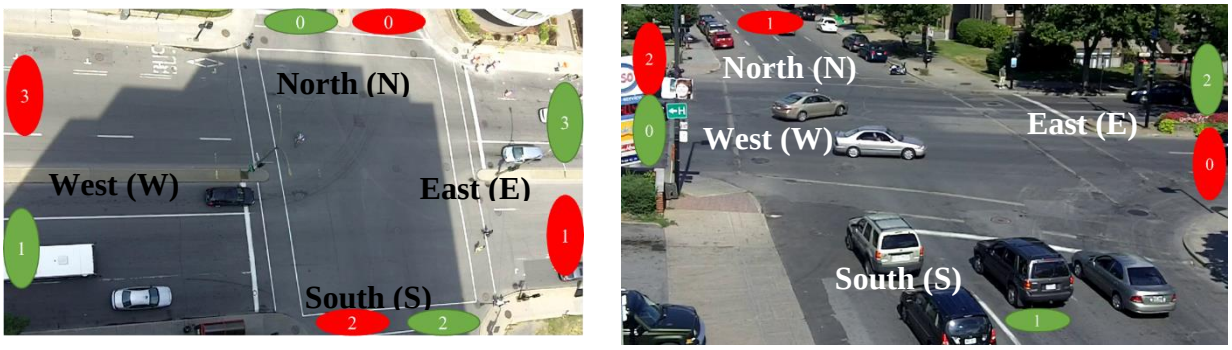
Figure 6.1(a) shows a sketch of entry and exit zones detected as POIs and used to specify the different movements in the intersection. Each route is defined by an entry and exit zone, for example, route (2,0) means that the road user enters scene through zone number 2 and exits from zone number 0 (routes can also be described with respect to geographic directions such as SN). In this case, left turn movements are represented by the following routes (0,1), (1,0), (2,3), and (3,2). Correspondingly, opposite through movement routes are (2,0) , (3,3), (0,1), and (1,1). Defining these routes helps to identify the LTOD interactions to be analyzed separately from the other types of interaction. There are four types of LTOD interactions at this intersection: information about the different types of interactions in terms of the traffic volumes of each involved movement, the numbers of interactions and the corresponding numbers of collisions is summarized in Table 6.1It can be noticed that the volume of left turn vehicles is a major variable that influences the number of interactions. In addition, there is an agreement between the number of interactions and the number of collisions for each type.

Table 6.1: Interactions descriptive and corresponding collision in Guy Intersection

Types of Interaction (ID)	**Left Turn Directions**		**Through Direction**		**Interactions Size/ hr**	**Collision Size (3years)**
	Route	**Volume/hr**	**Route**	**Volume/hr**		
INT1	(1,0)* WN	286	(3,3) EW	478	570	7
INT2	(3,2) ES	76	(1,1) WE	644	203	5
INT3	(2,3) SW	52	(0,2) NS	194	49	2
INT4	(0,1)** NE	27	(2,0) SN	204	27	1

* Route (1,0) is a protected /permitted left turn signal.

**Route (0,1) is a movement limited to authorized vehicles and taxis.



(a) Guy Intersection (b) Atwater Intersection

Figure 6.1: Sketch of locations of entry and exit zones learnt based on POIs (red ellipses represent entry zones while green ellipses represent exit zones)

6.2.2 **Atwater Intersection**

As described in Chapter 3, this intersection has a low angle FOV which is expected to affect the accuracy of safety indicators. Hence, this case study is limited to the analysis of the influence of road users' geometric representations on safety indicators. At this intersection, only one type of LTOD interaction exists since one street is unidirectional and 106 interactions were detected. As shown in Figure 6.1 (b), the routes of LT and OD are respectively WN (0,1) and EW (2,2). The left turn movement in this intersection is permitted.

6.2.3 **Kentucky Dataset**

This dataset was collected at one signalized intersection in Kentucky between August 16th 2001 and May 31st 2006. This dataset is described in detail in (Green, Agent, & Pigman, 2005) and it has two subsets of video recordings: one labeled "miss" corresponding to conflicts and the other is labeled "incident" corresponds to collisions. The dataset contains 295 interactions, split into 82 collision cases and 213 conflict cases. The definition of conflicts used by the people who collected and sorted the data is unknown: a visual review confirms that most match the accepted definition, but they will be referred to as interactions without a collision in this work. For each interaction, there is a video sequence lasting 10 s. The quality of the video is very low which makes the detection and tracking very challenging. However, this dataset has already been used in past studies (Saunier, Sayed, & Ismail, 2010) to compute several safety indicator. The definition of interaction

and the computing of indicators are compatible with our definition and method: our focus[7] in this thesis is on the use of the computed safety indicators to find similarities between interactions with and without collision.

6.3 Influence of Geometric Representation on Surrogate Safety Measure

PET is the selected surrogate safety measure to be calculated using four different automated measurement methods. The difference between each method is the road users' geometric representation. These methods are classified as: [1] CBM, [2] CirBM, [3] FBCH, and [4] FMAB. These methods are investigated in the case of LTOD interactions for two case studies: the Guy intersection which has an almost overhead view of the intersection, and the Atwater intersection which has a low angle view that can affect the results. Before proceeding to the analyses, some points should be mentioned:

- The extracted interactions contain complete and incomplete trajectories, it is expected that some trajectories will not cross each other,
- In some cases, an incomplete trajectory ends just before intersecting the other trajectory, in which case CBM cannot compute PET but CirBM can compute it if the distance between the trajectory end and the other trajectory is less than the threshold. Moreover, the end of incomplete trajectories mainly misses some features which makes it impossible to compute the PET using FBCH and FMAB methods.

Consequently, the comparison between the four methods is investigated on the common interactions for which the two interacting trajectories intersect each other and PET is computed by the four methods. To the end of thesis, a term "encounter" will be used as a substitute for an interaction that meets some criteria. For example in this section, an encounter is defined as an interaction in which PET is computed.

[7] The influence of motion prediction methods were investigated using this dataset in our previous papers (Mohamed & Saunier, 2013a; Mohamed & Saunier, 2013c)and it will not be presented in this thesis. Also, analysis of interaction similarities was partially presented in (Mohamed & Saunier, 2013b; Saunier & Mohamed, 2014)

6.3.1 **Guy Intersection**

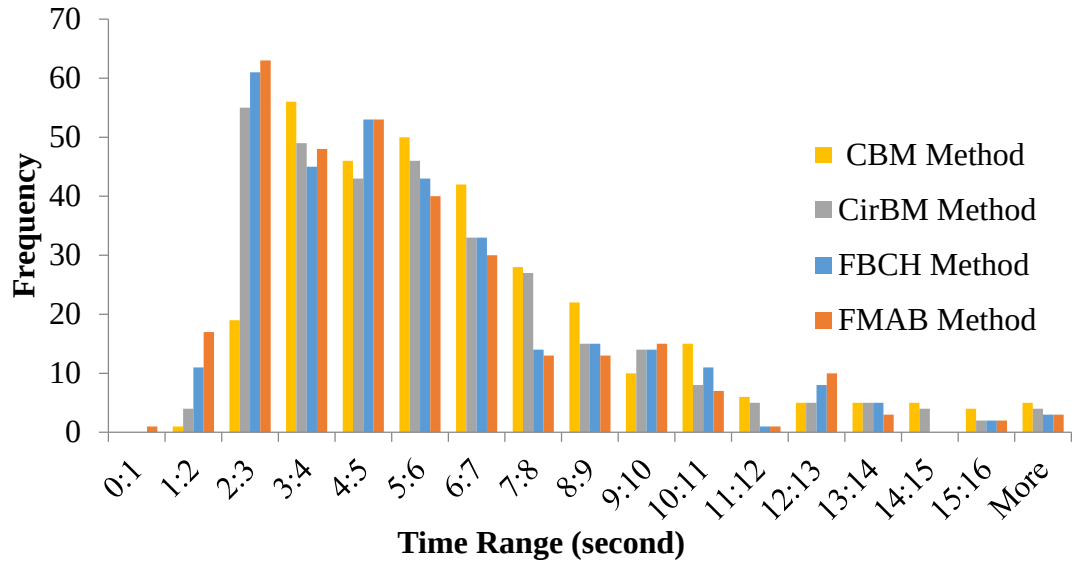
Among the extracted 849 LTOD interactions, only 319 encounters where PET can be computed for all geometric representations are detected. Table 6.2shows the number of encounters for each method and a comparison of the different methods. The comparison is based on two main criteria: the first criterion is the computation time based on the average of 10 runs and the second is some basic descriptive statistics (mean, standard deviation (s. d.), and median) of the PET distributions. As expected, the number of encounters is approximately similar for CBM and CirBM and for FBCH and FMAB. It is notable that CBM is a very fast method and is not even comparable with other methods. For the other methods, as expected, the computational time increases with increasing complexity of the geometric shape. For descriptive statistics, PET using CBM has the largest average value and the average values decrease with increasing complexity. The largest difference between median PETs reaches 1.2 s between CBM and FMAB.

Table 6.2: Comparison between PET measurement methods for the example dataset

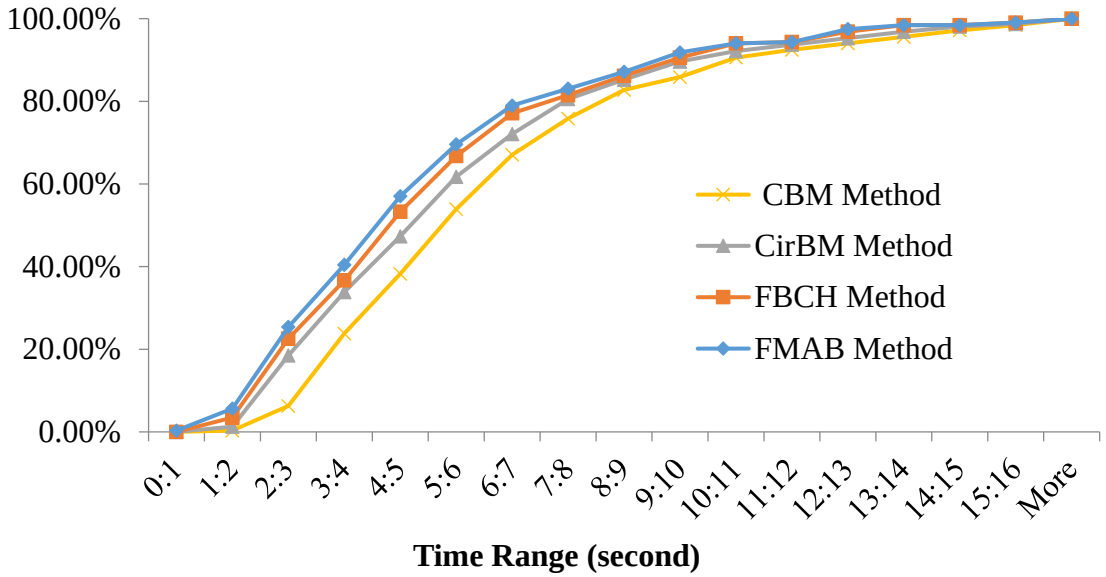
Method	**Number of encounters**	**Computation time (sec.)**	**Descriptive statistics**		
			Mean (sec.)	**s.d. (sec.)**	**Median (sec.)**
CBM	338	0.78	5.50	3.26	4.83
CirBM	341	37.46	4.88	3.22	4.23
FBCH	321	187.78	4.51	3.08	3.80
FMAB	323	196.55	4.30	3.07	3.63

The results for the measurement methods, histograms and Cumulative Distribution Function (CDF) of the PET distributions, are presented in Figure 6.2. The analysis shows that PET computed with the CBM method has the largest frequency in the [3:4[s range and only one measurement is found in the range of [1:2[s. This is not the case for other methods where there are some cases having lower PET values (range [1:2[s), there are 4, 11 , 17 cases for CirMB, FBCH, and FMAB respectively. In addition, one PET case in the range [0:1[is found using FMAB. This finding confirms that complexity of geometric shapes has an influence on the safety of interactions. In Figure 6.2 (a), PET values are shifted towards the smallest PET values in the following orders: FMAB, FCHM, CirMB, and CBM. CBM evaluates the studied interactions as the safest among other methods with a significant difference. Conversely, other methods have a slight shift in CDF

with increasing complexity. In the end, geometric shapes have a significant influence on the resulting safety evaluation.



(a) PET Distribution



(b) PET comulative distribution

Figure 6.2: Comparison between different geometric representation methods on PET.

To evaluate the accuracy of each method, the PETs computed by each method are compared with actual PET values. The actual PET values are estimated manually using the following procedure:

1) record the frame (instant) when first road user leaves the potential crossing zone,
2) record a second frame when the second road user arrives at the same area,
3) the actual PET is the difference between those two recorded instants.

40 interactions with PET values less than 1.5 s are selected based on measured PET using FBCH method. The root mean squared error (RMSE) and correlation analysis are computed to evaluate the accuracy of each method with respect to manual measurement. The RMSE is computed as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\hat{Y}_i - Y_i\right)^2}$$

Where n is the number of studied interactions, $\hat{Y}_i$ is the measured PET value using one of four measurement method, and Y_i is the actual value (estimated manually)

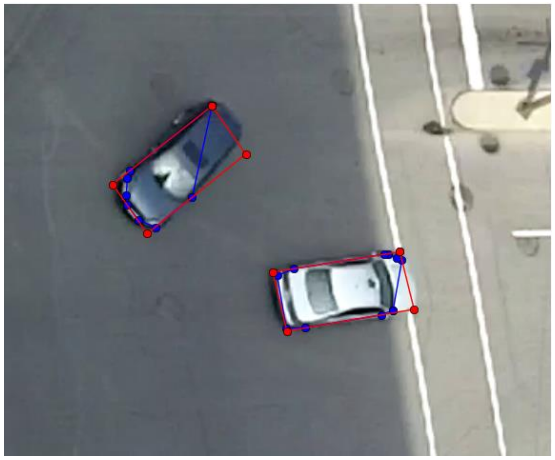
The RMSE values are respectively 1.22, 0.66, 0.25, 0.26 s using CBM, CirBM, FBCH, and FMAB. It is notable that the RMSE values of the FBCH and FMAB methods are very close and smaller than other methods. The correlation between the five measurement methods is performed and summarized in Table 6.3. As expected, there is high correlation between the FMAB and FBCH methods, and the CBM and CirBM methods. The manual measurements are highly correlated with FBCH, follow by FMAB. The overestimation of road user volume by the bounding box in FMAB leads, in some cases, to significant errors for PET estimation and to a weaker correlation with manual measurements.

Table 6.3: Correlations between the four methods and the manual measurements

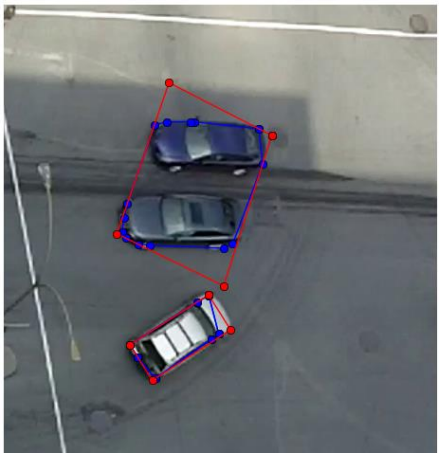
	CBM	**CirBM**	**FBCH**	**FMAB**	**Manual**
CBM	1.000				
CirBM	**0.936**	1.000			
FBCH	0.129	0.132	1.000		
FMAB	-0.228	-0.148	**0.742**	1.000	
Manual	0.301	0.317	**0.816**	0.538	1.000

Figure 6.3 shows specific examples of the FBCH and FMAB representations of some vehicles and the measurements of PET for the five methods under each picture. In the figure, blue boundaries

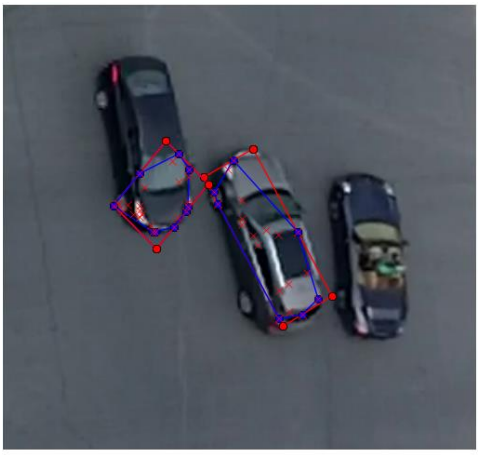
represent the FBCH volume while red boundaries represent the FMAB volume. Each interaction will be identified as I + sequence number of interaction (e.g. I345). In I666, two vehicles cross each other: vehicle outline is successfully estimated using FMAB while it is incompletely represented using FBCH, i.e. the use of bounding box corrects the issues created by the missing features in FBCH in this interaction. As a result, the FMAB method estimates PET equal to the manual measurement. I122 is an example of an over grouping error where two vehicles are assigned as one vehicle. FMAB estimates the boundary of the two vehicles wrongly and leads to measuring a lower PET value. For this example, FBCH estimates PET more closely to the manual estimation as the actual boundary is represented. The other two methods measure much larger PET values, more than twice the actual PET value. I597 and I598 include over-segmentation errors where one vehicle is split into two vehicles and accordingly two interactions are detected. In these examples, FBCH estimate PET better than the other methods. I112 and I668 show other road user types, a cyclist and a bus respectively. Results show the superiority of FBCH and FMAB to measure PET where these methods are able to consider the actual size of each road user. These examples show the superiority of FBCH and how it is less prone to tracking errors and miss-classification problems to estimate surrogate safety indicators. Moreover, it has the advantage of using all tracked features without any assumptions.



I666	0.90	0.73	1.03	1.53	0.73

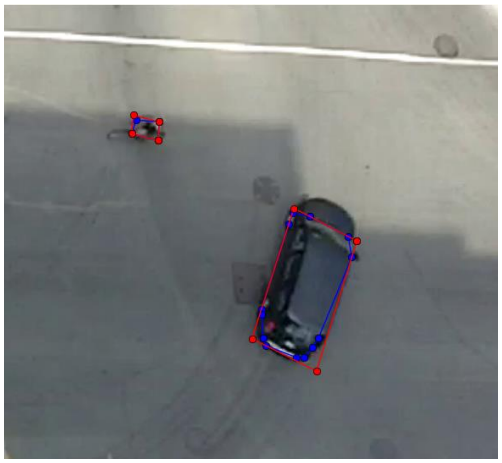


I122	1.40	0.43	2.50	3.33	1.27

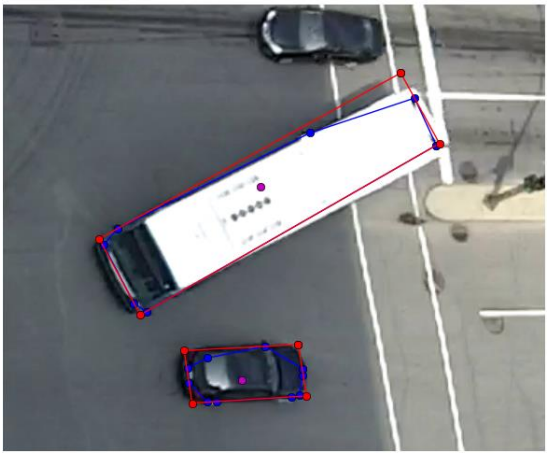


I597	1.03	0.00	1.67	2.70	0.80

I598	0.83	0.77	1.43	2.43	0.80



I112	1.40	1.30	1.53	2.10	1.30



I668	0.70	0.53	1.47	1.90	0.47

Figure 6.3: Examples of PET measurements (in the table under each image representing the interaction from the FBCH, FMAB, CirMB, CBM, and Manual methods)

6.3.2 Atwater Intersection

In this case study, only 57 encounters of the LTOD interactions are detected. The RMSE values are relatively higher than for the Guy Intersection and its values are as follows, 1.06, 0.81, 1.43, and 1.62 s for the CBM, CirBM, FBCH, and FMAB methods. CirBM has the lowest error among all the methods. Surprisingly, all methods are highly correlated with manual measurements. On the contrary to the findings of the previous section, FBCH and FMAB report large errors. This can be explained by the fact that the detected features for each vehicle represent the top and some sides of vehicles because of the camera low angle which results in larger estimated vehicle outlines (which is worse when projected in world coordinates). It should be mentioned that the manual measurement may not be very accurate because it is difficult to identify the crossing zones and consequently the instants at which the road users enter or leave the crossing zone.

6.4 Influence of Motion Prediction Methods on Surrogate Safety Measures

This experimental analysis relies on the LTOD interactions of the Guy intersection. Four methods for motion prediction are applied to predict collision points and crossing zones, and to compute the associated TTC and pPET indicators. These methods are CV, NA, and the application of the multi-level motion pattern learning sub-framework with two assumptions: 1) MPCS (using destination and constant speed) and (2) MPSP (using destination and temporal prototypes). The NA method requires setting a number of trajectories (N_1) (for sampling the control inputs) and the distributions of control inputs. They were chosen to be N_1=100 and the triangle distributions with 0 for mode and range [-2 m/s^2,2 m/s^2] for acceleration input, and range [-0.2 rad/s, 0.2 rad/s] for steering input. For MPCS, two main inputs are required: 1) using POIs to define entry and exit zones for each interacting trajectory, 2) a set of spatial prototypes to predict future motion. MPSP takes the same inputs, in addition to the temporal prototypes. Moreover, some parameters need to be defined such as the minimum length of partial trajectory to match with prototypes (the selected length is 30 instants). Samples of the safety indicators, the collision points and the crossing zones are plotted in Figure 6.5 to Figure 6.8 for the studied interactions. The names of the motion prediction methods are abbreviated to avoid overlapping with the plots. As this section focuses on the influence of motion prediction methods on the computation of safety indicators, the road user is considered as

a circle and a point when measuring TTC and pPET respectively to simplify and accelerate the analysis.

6.4.1 Number of Measurements and Detected Encounters

In this section, an encounter is an interaction for which the surrogate safety indicators (TTC and pPET) can be computed. It is noteworthy that each motion prediction method detects a different number of encounters and shows a significant difference in the number of measurements for the TTC and pPET indicators. Table 6.4summarizes the number of encounters using different motion prediction methods. Prediction at constant velocity provides the smallest number of encounters for TTC and pPET, followed by MPCS, then the sampling of trajectories with normal adaptation and finally the method based on MPSP. For the number of measurements as shown in Figure 6.4, it is remarkable that CV and NA provide the smallest number of measurements where around 50 % of encounters have fewer than 5 measures over time. It was expected that motion prediction at constant velocity would provide the smallest number of measurements, which is a well-known shortcoming of that method (Laureshyn A. , 2010). The indicators computed with the NA method are a smoother version of the indicators computed with the CV method, with more measurements. On the other hand, MPCS and MPSP have only 5 to 10 % of encounters that have 4 measures or fewer over time. This important characteristic is associated with robustness as measurements over longer periods of time should help better characterize the interactions over time and in terms of their overall safety, while a small number of data points provides a limited picture and is more subject to noise.

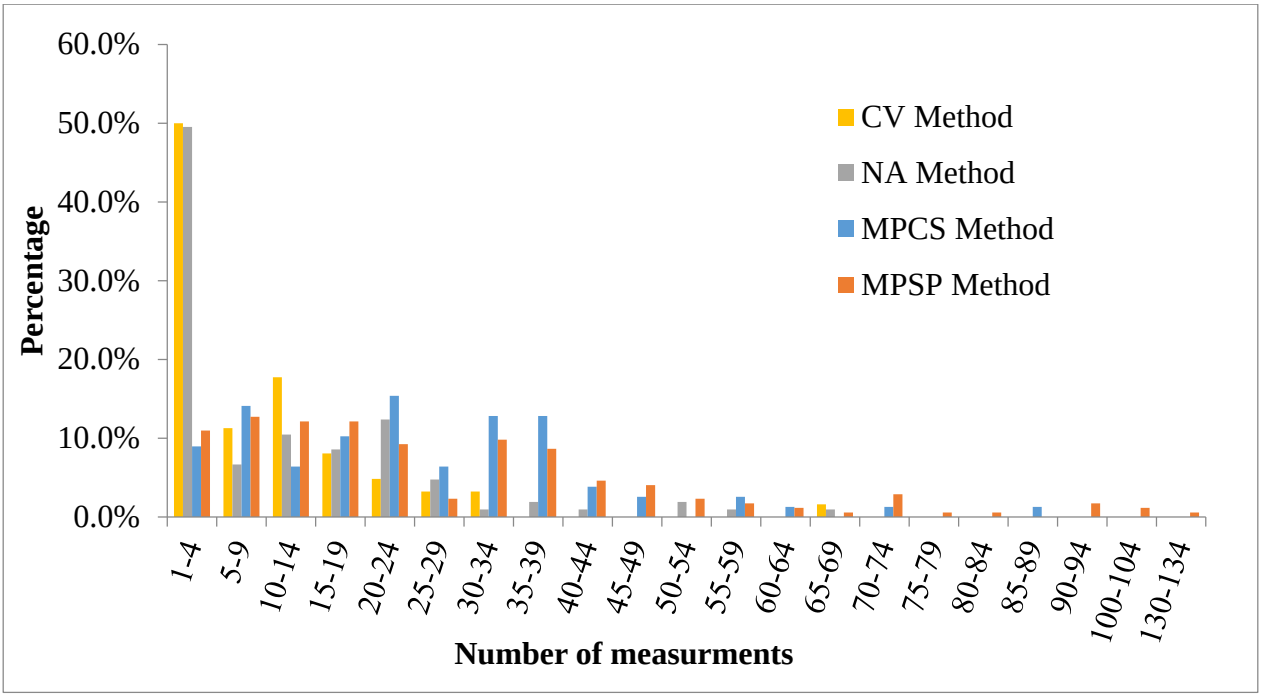


Figure 6.4: Distribution of the number of measurements per interaction based on the calculated TTC using different motion prediction method.

Table 6.4: Number of encounters for each motion prediction method based on computed TTC and pPET (time horizon is 150 frames (equivalent to 5 s))

	INTIDs	CV		NA		MPCS		MPSP	
		TTC	pPET	TTC	pPET	TTC	pPET	TTC	pPET
All encounters	INT1	43	58	76	97	36	82	123	214
	INT2	18	44	26	70	36	57	43	65
	INT3	1	3	3	10	5	17	5	20
	INT4	0	1	0	1	1	9	2	12
	Total	**62**	**106**	**105**	**178**	**78**	**165**	**173**	**311**
Encounters with more than 4 measures	INT1	21	32	36	51	34	81	109	204
	INT2	9	26	15	52	33	52	38	59
	INT3	1	2	2	4	3	16	5	19
	INT4	0	0	0	0	1	8	2	11
	Total	**31**	**60**	**53**	**107**	**71**	**157**	**154**	**293**

The collision points (CPs) and crossing zones (CZs) are presented respectively in Figure 6.5 and Figure 6.6. Comparing CPs and CZs within the same prediction method, it is clear that the spatial

distributions of CPs and CZs are similar, but the number of CZs is significantly larger than the number of CPs. Logically, CPs correspond to less safe events.

Furthermore, the distribution of CPs and CZs is compared between different motion prediction methods. As expected, the number of CPs and CZs predicted by the CV methods is smaller than for the other methods and sparser in the scene, some of them located in areas that cannot be realistically expected to be reached in the studied interactions. For example, a turning vehicle is very unlikely to move in a wrong direction and collide with an opposite direction vehicle. Similarly, the points and zones predicted by the NA method are more distributed than the ones predicted by the CV method. This is also expected since normal adaptation simulates small deviations around a trajectory at CV that are compensating each other since positive and negative values of control inputs can be drawn with equal probabilities. On the contrary, the estimated CPs and CZs using MPMs are concentrated around the expected trajectory intersections.

The CZs and CPs corresponding to TTC and pPET values less than 1.5 s are presented in the right column of Figure 6.5and Figure 6.6. Clearly, the number of CPs is relatively small for all methods: MPSP method reports the highest number among other methods which the majority of these points is belong to INT1. Moreover, kinematic methods compute CPs for TTC ≤ 1.5 less sparse than using all CPs and these CPs concentrated in the trajectory intersections belonging to INT1. On the other hand, CZs for pPET ≤ 1.5 have a significant large number compared to CPs for all methods except CV method. The spatial distribution of CZs based on kinematic methods is sparse over the scene (similar finding using all CZs) which is difficult to identify the locations and/or interaction types that have safety problem. However, MPM methods are clearly identified that there is a safety problem in INT1 and INT2 and in which locations having frequent trajectory intersections. This discussion encourages the usage of MPMs over kinematic methods, especially for interactions involving turning movements.

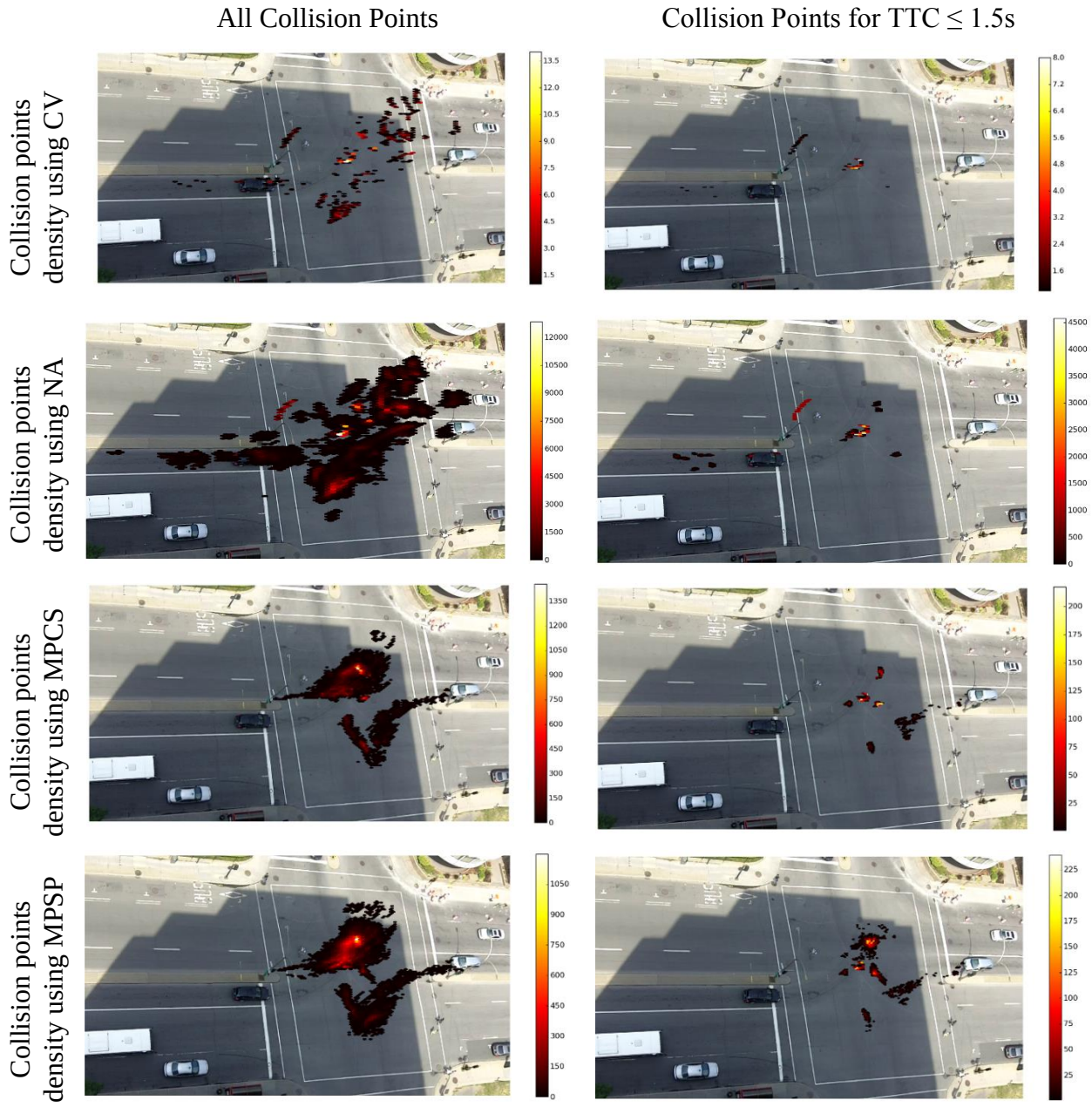


Figure 6.5: Maps of collision point's density for LTOD interactions using different motion prediction methods. Left maps represent all computed collision points while right maps show collision points corresponding to TTC value below 1.5 s.

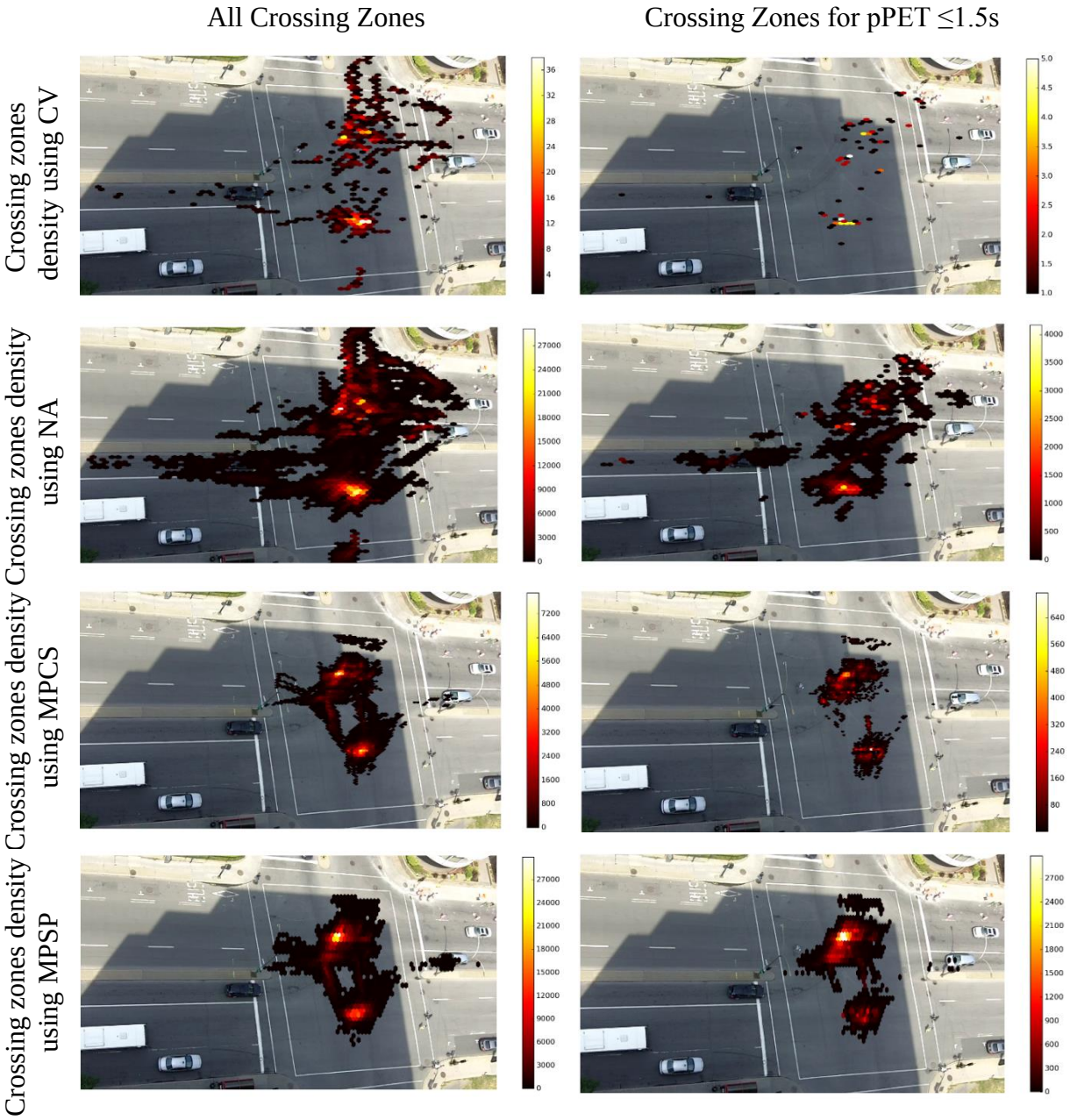


Figure 6.6: Maps of computed crossing zones density for LTOD interactions using different motion prediction methods. Left maps represent all computed crossing zones while right maps show crossing zones corresponding to pPET below 1.5 s.

6.4.2 **Safety Indicator Calculation**

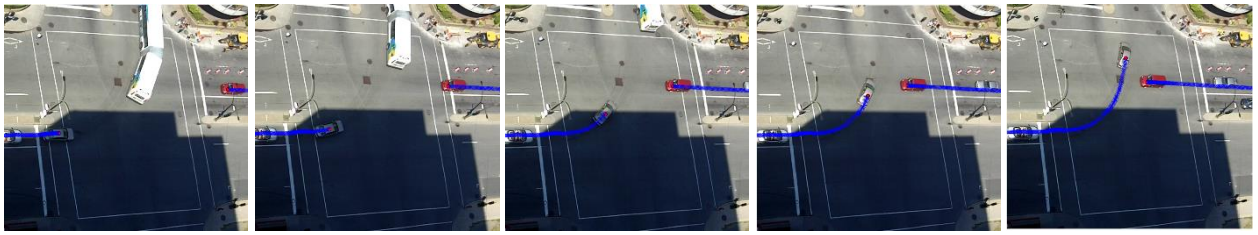
For the sake of testing the performance of each motion prediction method for surrogate safety analysis, two examples of encounter are studied in details. To understand the driver behaviour during the interaction interval, simple and temporal indicators are analyzed in each encounter.

Figure 6.7 and Figure 6.8 present two interactions (numbers 689 & 529) collected around noon between Left Turn (LT) and Opposite Direction (OD) vehicles. Based on manual PET measurements, interaction 529 is safer than interaction 689 where PET value for the former is 1.06 s and for the latter is 0.6 s.

For the interaction 529 shown in Figure 6.7, the speed profiles exhibit some complex changes over time during the interaction interval. For example, the LT vehicle is accelerating till it reaches a speed of 21 km/hr (a relatively small speed value), then it suddenly starts to decelerate at the same time that the OD vehicle decelerates as well, which indicates that the two drivers noticed the dangerous situation. Both of the LT and OD vehicles performed a braking action reaching up to -8 m/sec^2. In this interaction, the four motion prediction methods are able to report values for the safety indicators but with different numbers of measurements. Notably, MPM methods predict safety indicator values earlier than the kinematic method (starting around 1.5 s after the beginning of the interaction). Some TTC values are similar in all methods when the two vehicles are very close to the observed trajectory intersection in which case the possible predicted trajectories are basically limited to the ones generated by CV. The TTC using the CV and NA methods can be computed only when the vehicles start braking in this interaction. As a result, TTC using MPM methods provides a better, more complete picture of the interaction that helps to understand the interaction process. On the contrary, pPET values are further apart for the MPM and the kinematic methods, especially before the vehicles reach the trajectory intersection, because of the prediction of unrealistic future motion for the left turn movement. At the end of the interaction, it is expected that the pPET values are very close to PET value based on the CBM method, since one road user actual trajectory already crosses the other road user's actual trajectory. Since the LT vehicle leaves the CZ at an instant between 2685 and 2686, pPET is computed between the predicted trajectories for OD and the actual trajectory of LT vehicle. The evolution of pPET profile is continued till the instant when both actual trajectories intersect or the last instant of interaction interval. In this interaction, the pPET profile ends at the last instant in the interaction interval which is 9 frames (0.3 sec.) earlier than when the actual trajectories intersect. Because of the road user’s representation as a point, it is expected that the measured pPET values are more than the actual values by about 1.25 s (the RMSE computed in the influence of geometric representative section (section 6.2)). As this interaction is one of the manually measured PET, the exact error of measured

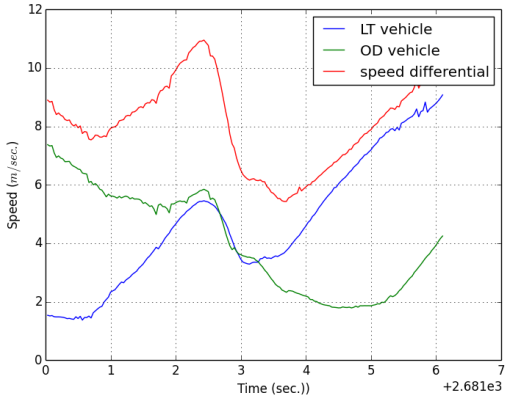
PET is known to be 0.93 s between the CBM and manual methods. It should affect the computation of pPET with approximately the same error.

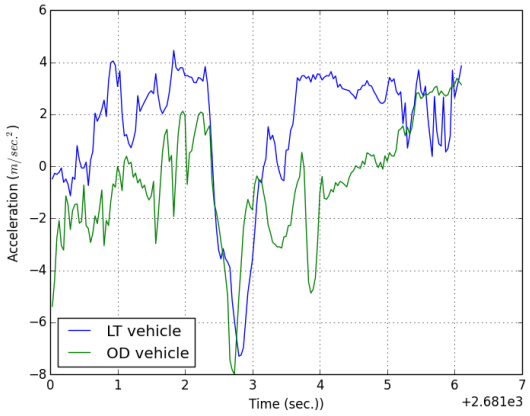
In the interaction 689 shown in Figure 6.8, the drivers' speed profiles are relatively normal. The LT vehicle moves at constant speed (15 km/hr) then starts accelerating when there is still no collision course and no TTC can be computed (around 1 s). The OD vehicle has a larger speed (50 km/hr) at the onset of the interaction and decelerates till it reaches a speed of 25 km/hr. This is a typical behaviour of LTOD interactions where LT vehicles wait until OD vehicles passes or until the LT driver estimates the gap is large enough to perform the left turn. In this example, it is noticeable that OD vehicle makes a braking action at the instant 964 with a value of – 7 m/sec^2. At this instant, a critical TTC is reported using MPM methods which the kinematic methods fail to report. The pPET values are measured using the four prediction methods. Similar to the previous interaction, the pPET values are larger before the instant at which the actual trajectories intersect. The pPET profiles are computed till they reach the PET value and all methods coincide then at a value equal to 1.33 sec (as PET measured using CBM method) in this interaction.



2682 2683 2684 2685 2686

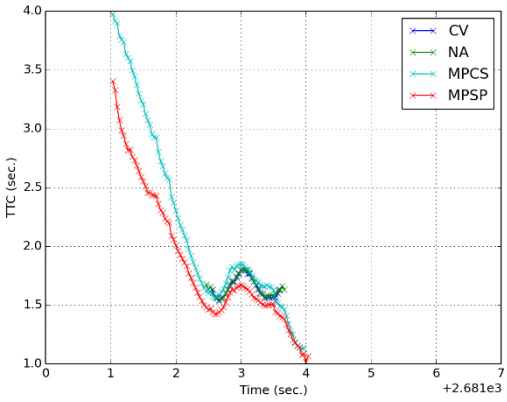
(a) Interaction snapshots (instants): the blue crosses are the past positions

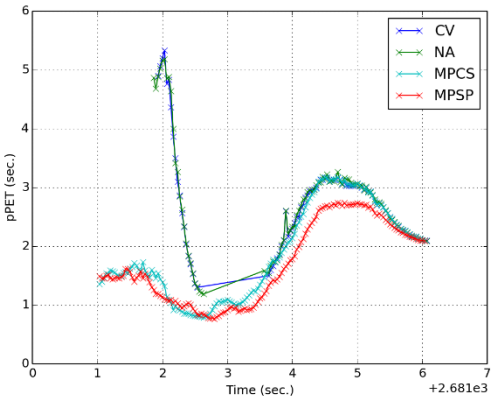




i)Speed profiles ii) Acceleration profiles

(b) Kinematic indicator profiles

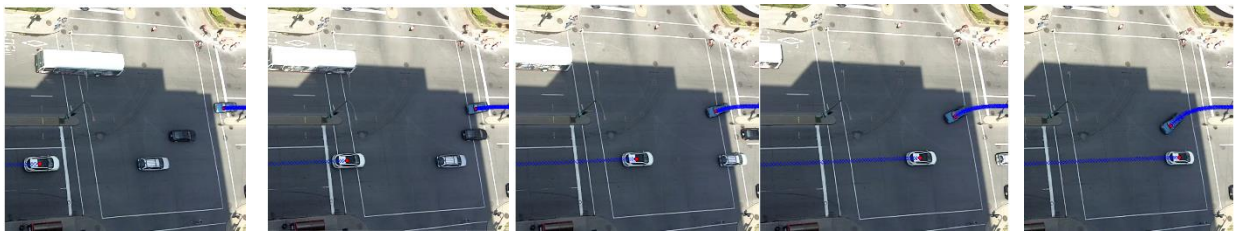




i)TTC profiles ii) pPET profiles

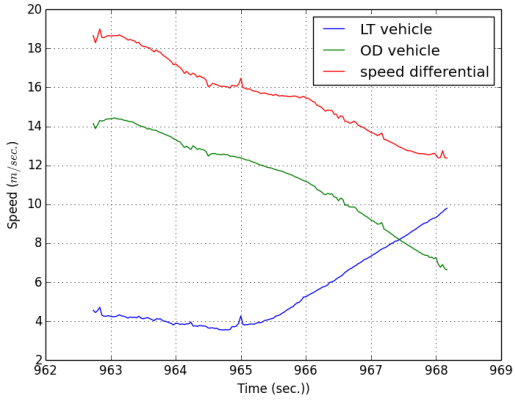
(c) Safety indicator profiles

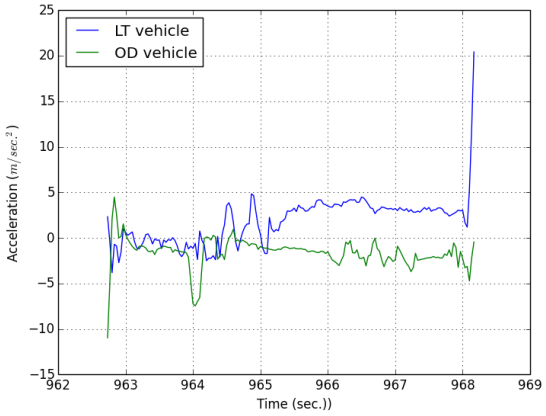
Figure 6.7**:** Example of LTOD interaction (interaction 529).



964 964.5 965 965.5 966

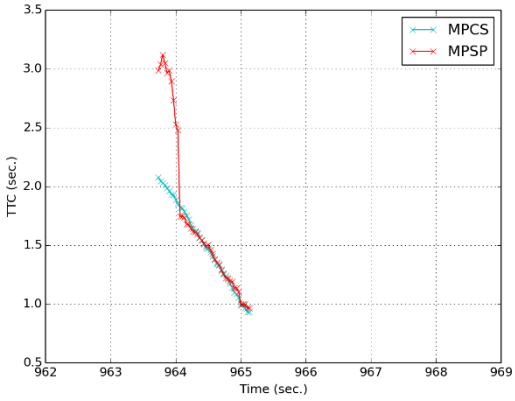
(a) Interaction snapshots (instants): the blue crosses are the past positions

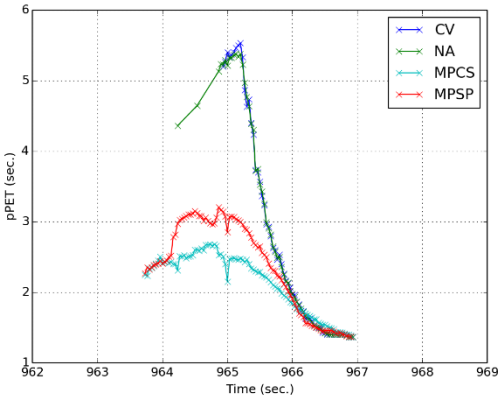




a)Speed profiles b) Acceleration profiles

(b) Kinematic indicator profiles





i)TTC profiles ii) pPET profiles

(c) Safety indicator profiles

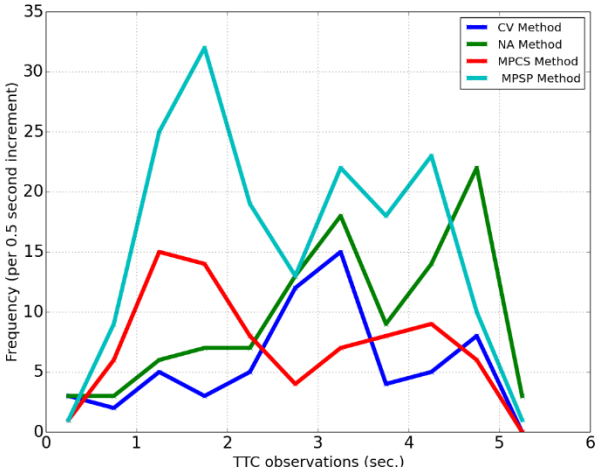
Figure 6.8**:** Example of LTOD interaction (interaction 689)

6.5 Analysis of Safety Indicator Distributions

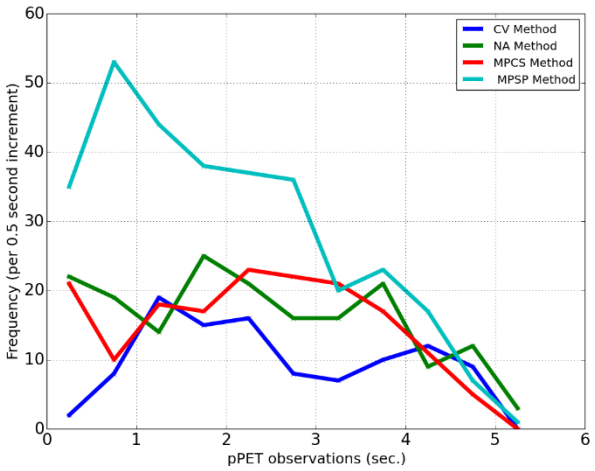
6.5.1 Aggregation by Site

The safety of the studied LTOD interactions at the Guy intersection is evaluated by analyzing the distributions of safety indicators such as minimum TTC (TTC_{min}) and minimum pPET ($pPET_{min}$) computed by the four motion prediction methods. The distributions are presented in Figure 6.9 where it is clear at the first sight that each motion prediction method has a different evaluation of safety. For example, as shown in Figure 6.9 (a), TTC_{min} distributions have a peak around 1.5 s for the MPCS and MPSP methods. In addition, another peak of TTC_{min} measurements of more than 3 s is found in the case of the MPSP method. However, an almost flat TTC distribution, with a small peak found around 3.5 s, is noticed in case of CV and NA methods. This leads to different evaluations of the same interaction depending on the motion prediction method. This finding shows the importance of the motion prediction method in safety estimation.

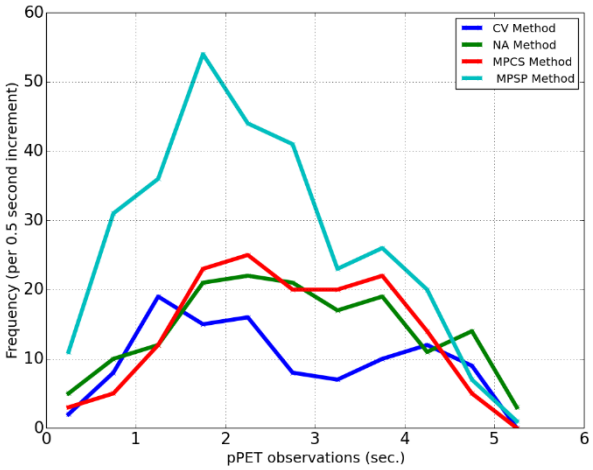
The pPET should be complementary to the TTC by construction, however, its interpretation in our cases studies is not simple. It is clarified earlier that the CV method can compute either pPET or TTC at any instant as only one trajectory is predicted. Probabilistic prediction methods, NA and MPM methods, may calculate both TTC and pPET values at any instant where many trajectories are predicted. It is noteworthy and expected in some cases that pPET values are very small in case of computed TTC. To analyze $pPET_{min}$ distributions, two assumptions are investigated: 1) considering all pPET values (see Figure 6.9 (b)) and 2) ignoring pPET at instants that have TTC values (see Figure 6.9 (c)). The main difference between the two distributions is the significant reduction in the frequencies of lower values (in the range of 0 to 1.0 s) in the NA and MPM methods. The MPSP method reports the distribution with the highest $pPET_{min}$ with a peak around 1.5 s. Although pPET profiles have a potential usefulness of understanding the interaction process in the absence of detected CPs and the behaviour of interacting road users till their trajectories cross each other, the integration of pPET and TTC is still an open question that needs further investigation.



(a) TTC distributions



(b) pPET distributions considering all values



(c) pPET distributions ignoring values at instants that have TTC values

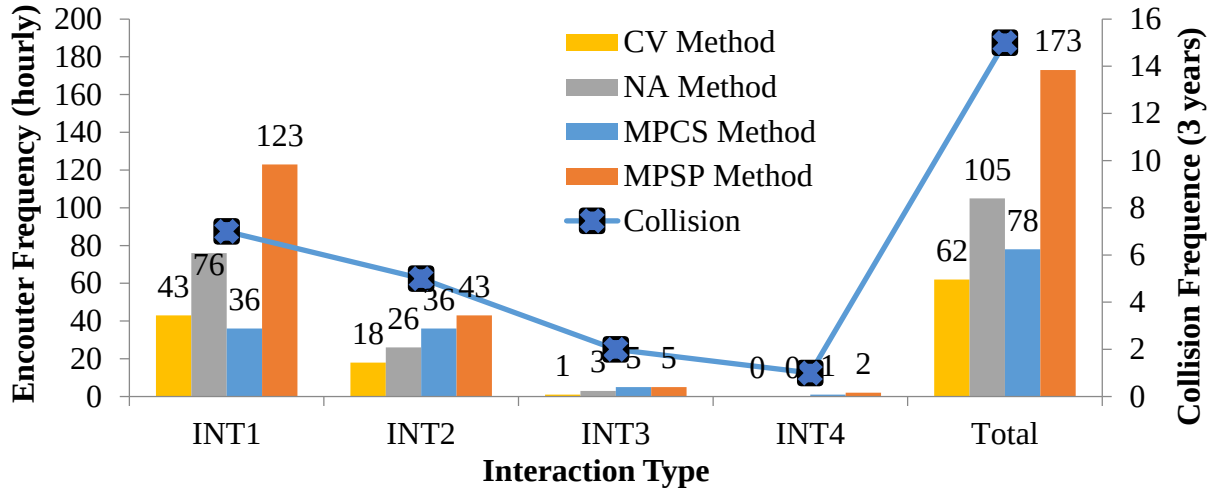
Figure 6.9: Distributions of TTC_{min} and $pPET_{min}$ observations using different motion prediction methods

6.5.2 **Aggregation by Interaction Type**

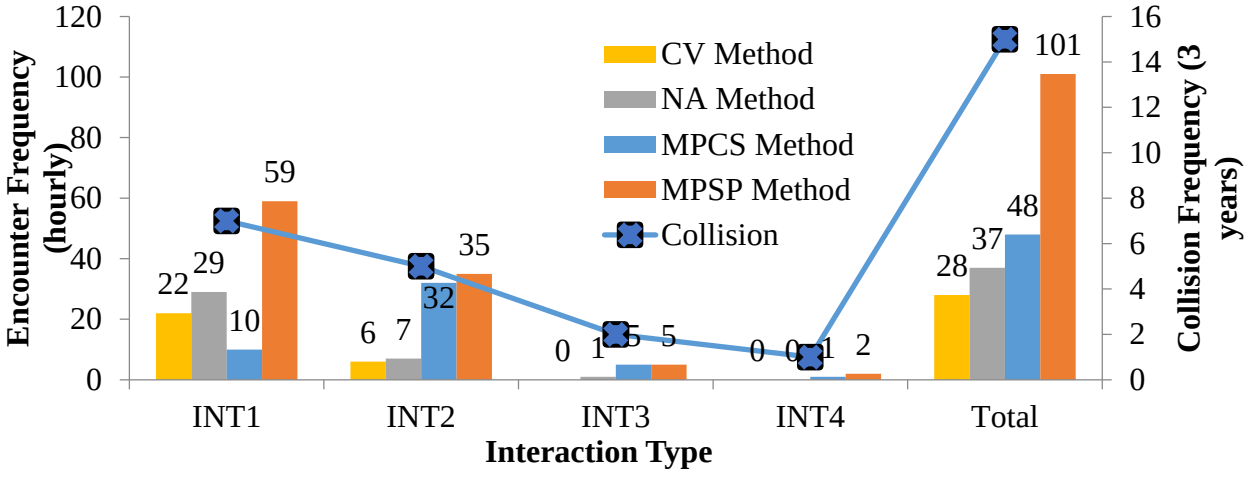
For each motion prediction method, the collected TTC_{min} were classified based on the four interaction types in the Guy intersection. The interaction types correspond to the four types of LTOD interactions in a four legged intersection. Three thresholds are selected i.e. 5, 3, and 1.5 s. For each threshold, the number of encounters based on each prediction method and the collision frequency are compared (see Figure 6.10). Considering a 5-s threshold, all prediction methods agree that INT1 and INT2 have more frequent interactions than the INT3, INT4 categories which is similar to collision frequencies. However, the MPCS method reports an equal interaction frequency for both INT1 and INT2. Considering a 3-s threshold, all methods except MPCS still maintains an agreement with the collision frequency: the MPCS method reports that INT2 has more

frequent interactions than INT1. When using a 1.5 s threshold, the surprising results are the following:

- The MPM methods report that INT2 has the largest number of encounters, however, the kinematic method reports only 1 interaction in this interaction type.
- For INT1, the MPSP method reports a large number of encounters compared to the MPCS method and has a small difference with the number of encounters identified by the kinematic method.

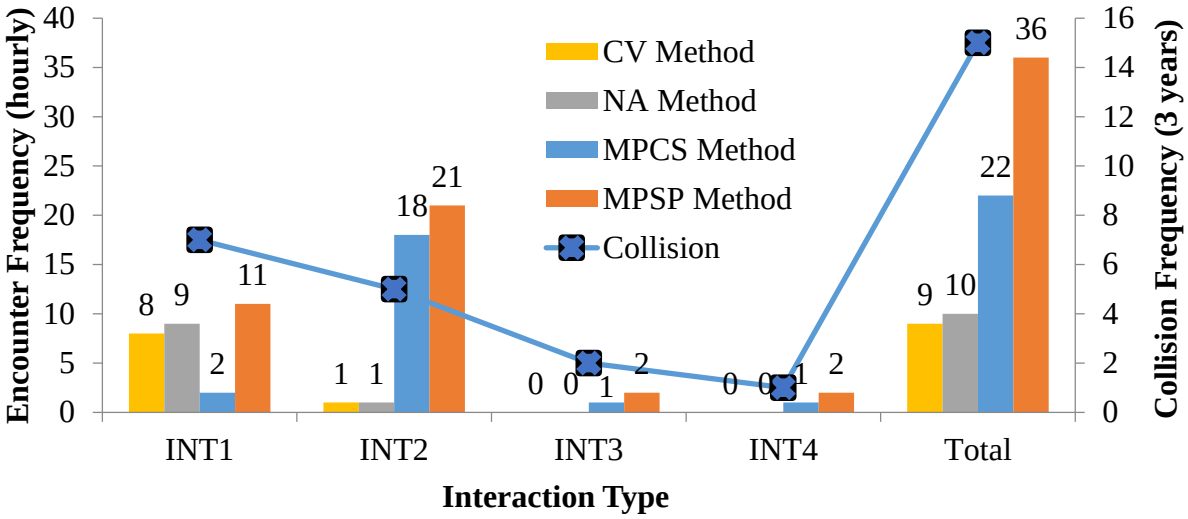


(a) TTC_{min} value <= 5 s



(b) TTC_{min} value <= 3 s

Figure 6.10: Comparison of the frequency of encounters (with different TTC_{min} thresholds) and collisions for different motion prediction methods and types of interaction



(c) TTC_{min} value <= 1.5 s

Figure 6.10: Comparison of the frequency of encounters (with different TTCmin thresholds) and collisions for different motion prediction methods and types of interaction (cont'd)

As summarized in Table 6.5, the correlation between the number of encounters and collision frequency is analyzed based on Pearson linear correlation coefficient. It is found that a high correlation exists for all methods when the TTC_{min} threshold is 5 s. The MPCS method has the lower correlation values for thresholds 3 and 1.5 s. On the other hand, a Spearman's rank correlation is analyzed based on the ranks of interaction types and collisions. In this case, the MPSP method reports high correlations for thresholds 5 and 3 s and lower correlation for the 1.5 s threshold. MPCS reports the lowest correlation values among all methods. Surprisingly, kinematic methods show high correlation values for all thresholds.

Table 6.5: The correlation analysis between the frequency and the rank of encounters and collisions for different thresholds and using TTC_{min}

	Spearman's Rank Correlation			Pearson correlation		
	5 s	3 s	1.5 s	5 s	3 s	1.5 s
CV Method	1	0.949	0.949	0.966	0.920	0.854
NA Method	1	1	0.949	0.945	0.906	0.847
MPCS Method	0.949	0.800	0.738	0.953	0.534	0.355
MPSP Method	1	1	0.738	0.944	0.992	0.708

Although the previous findings rely only on one example, it shows that the proposed method based on the MPSP method is a promising tool for surrogate safety analysis. In addition, the choice of motion prediction method could affect the safety diagnosis of the studied site or the comparison

between different sites. The choice of threshold to estimate safety proves to be a challenging task that affects the safety diagnosis.

6.6 Clustering of Safety Indicators Profiles

Time-series (profiles) of TTC are selected for the clustering analysis. A threshold equal to 0.2 s is chosen by trial and error to match the profiles using the aligned normalized similarity *SALCSS* with a finite δ. An additional criteria is added to remove very short indicators that do not contain much information (minimum length). In addition, the refinement algorithm is applied to small clusters with a minimum number of 8 profiles, including the prototype in each cluster. The clusters containing a smaller number of profiles are grouped as outliers and are examined separately. The parameters of the clustering algorithm for each case study are listed in Table 6.6.

Table 6.6: Parameters used in the computation of SALCSS and clustering algorithm (fps is frames per second)

	Threshold (s)	Minimum similarity	δ (frames)	Minimum length (frames)
Kentucky Dataset (frame rate 15 fps)	0.2	0.3	2	10
Guy Intersection (frame rate 29.97 fps)	0.2	0.4	5	15

6.6.1 **Kentucky Dataset**

The TTC profiles calculated in (Saunier, Sayed, & Ismail, 2010) are used to find the similarities between interaction with and without collision. TTC was computed based on motion pattern matching method. It must be noted that there are only 247 interactions for which TTC can be computed for at least 10 frames. As presented in (Saunier & Mohamed, 2014), the analysis of TTC indicator result a 4 clusters as shown in Figure 6.11: the numbers in each cluster title are in order the percentage of collision, and the number of interactions with collision and the number of all interactions in the cluster. The profiles of TTC contain a high level of noise because of the low quality of the data. As expected for collisions and conflicts, most TTC profiles decrease with time. Interestingly, the first two clusters look similar at first sight, however, the proportion of collisions in cluster 2 is consistent with the profile of its prototype indicator which falls at a seemingly constant rate as a function of time and reaches almost 0 s. On the contrary, there are few TTC

measures below 0.5 s or even 1 s in cluster 1. There is more variability in the rate of decrease at the beginning and most profiles increase again after reaching their minimum, which is consistent with a high proportion of interactions without a collision. Cluster 3 contains mostly collisions, with a higher rate of decrease than cluster 2 which explains why they are in different clusters. Finally, cluster 4 contains only one collision and has TTC values above 1.5 s. The results yield further credibility to the main hypothesis of surrogate safety analysis that some interactions without a collision have similar processes as collisions and could be used as predictors.

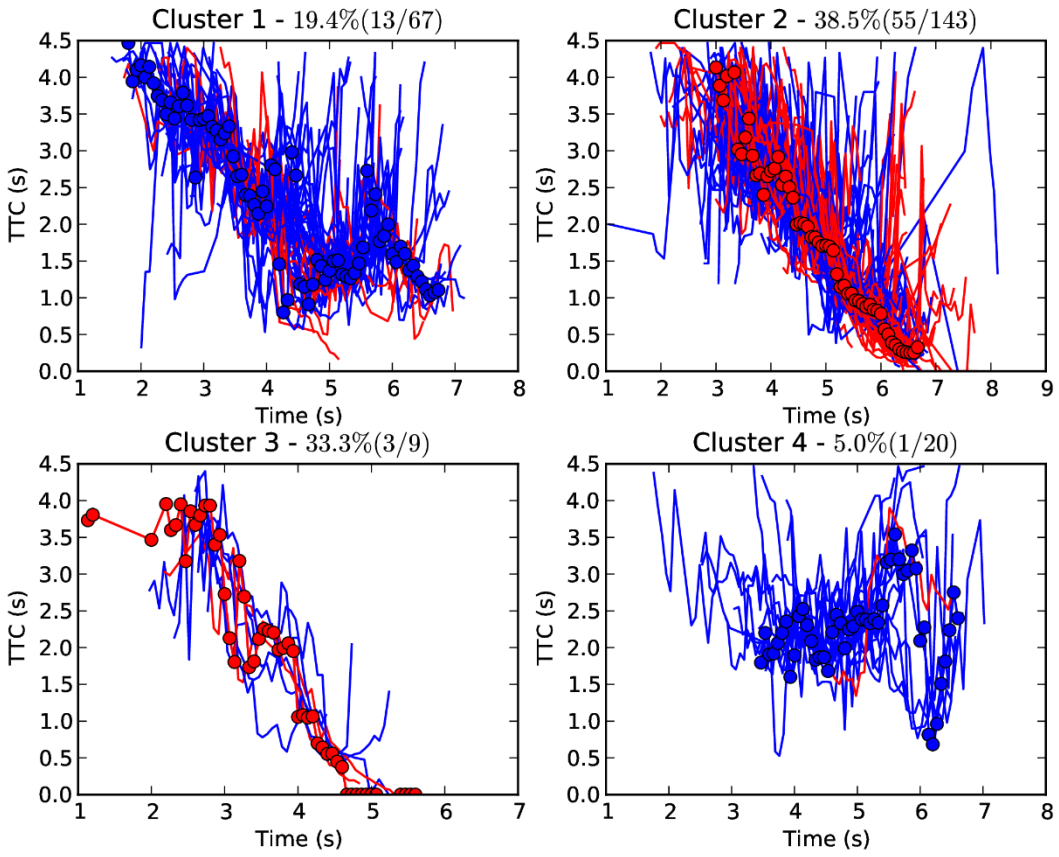


Figure 6.11: Clustering of the TTC profiles in the Kentucky dataset (Saunier & Mohamed, 2014)

6.6.2 **Guy Intersection**

For the Guy intersection, the TTC computed using the MPSP method is studied. It must be noted that there are only 111 encounters (among total of 311) for which TTC can be computed for at least 15 frames (0.5 s). TTC profiles are classified into 5 clusters as shown in Figure 6.13: each cluster prototype is plotted in red. The numbers beside each cluster number represent the cluster size as a percentage of the total number of encounters and the number of encounters in the cluster out of the

111 encounters in the parentheses. It is clear that the TTC indicators are noisy but less than the ones in the Kentucky dataset. The profiles in cluster 4 are flat and have values between 4 and 5 s. In cluster 4, some noisy values are noted to reach 2.5 s, these values can over-estimate the unsafety of the interaction if measured by only one value. In cluster 1, TTC profiles decrease with time till they reach values equal to 3 s then increase. Cluster 2 is very interesting as it has the lowest TTC values and is steeper than others. Cluster 3 has a high level of noise in the prototype. Finally, Cluster 5 have a smaller share of all the interactions and smaller numbers of measurements around 1.5 s. A visual ranking of the profiles based on the proximity to collision would be as follows: cluster 2, cluster1, cluster 3, cluster4, and cluster 5. A further investigation is conducted by demonstrating the share of each interaction type in each cluster as presented in Figure 6.12. It is clear that INT1 has the highest number of detected interactions in which cluster 1 is the largest while INT2 has the majority of detected interactions belonging to cluster 2 (the least safe cluster). This finding may be similar, in some sense, with the previous general finding of the distribution analysis. However, it has several advantages: there is no need to aggregate the profiles as a single value, the analysis is less sensitive to noisy values (outliers) and there is no need to select different thresholds.

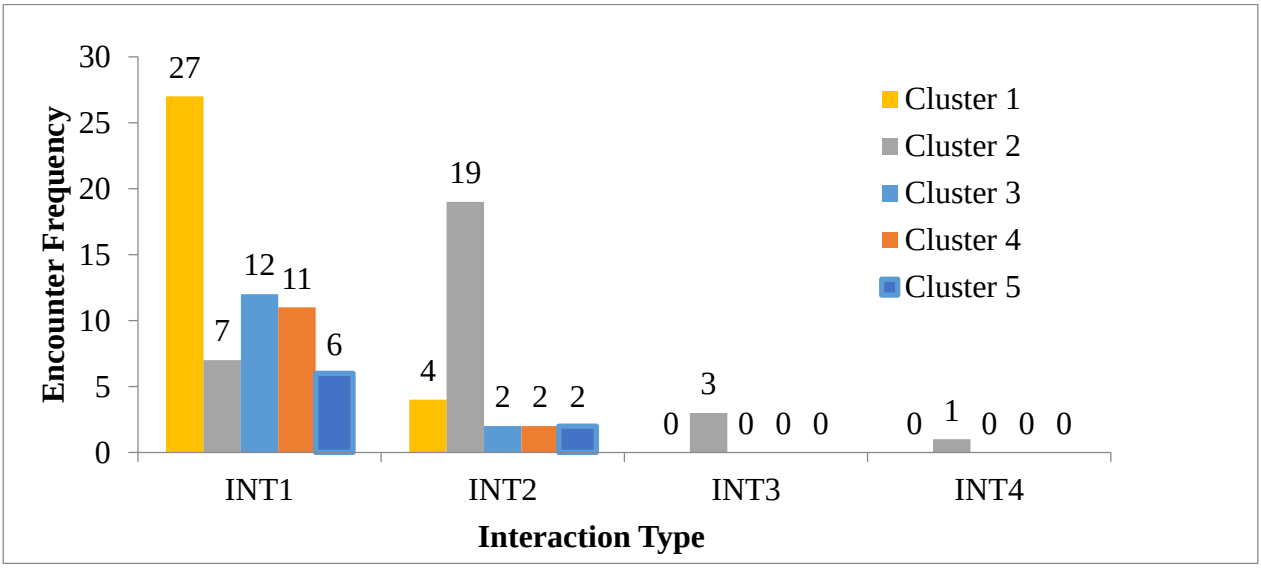


Figure 6.12: The distribution of each cluster of TTC per interaction type.

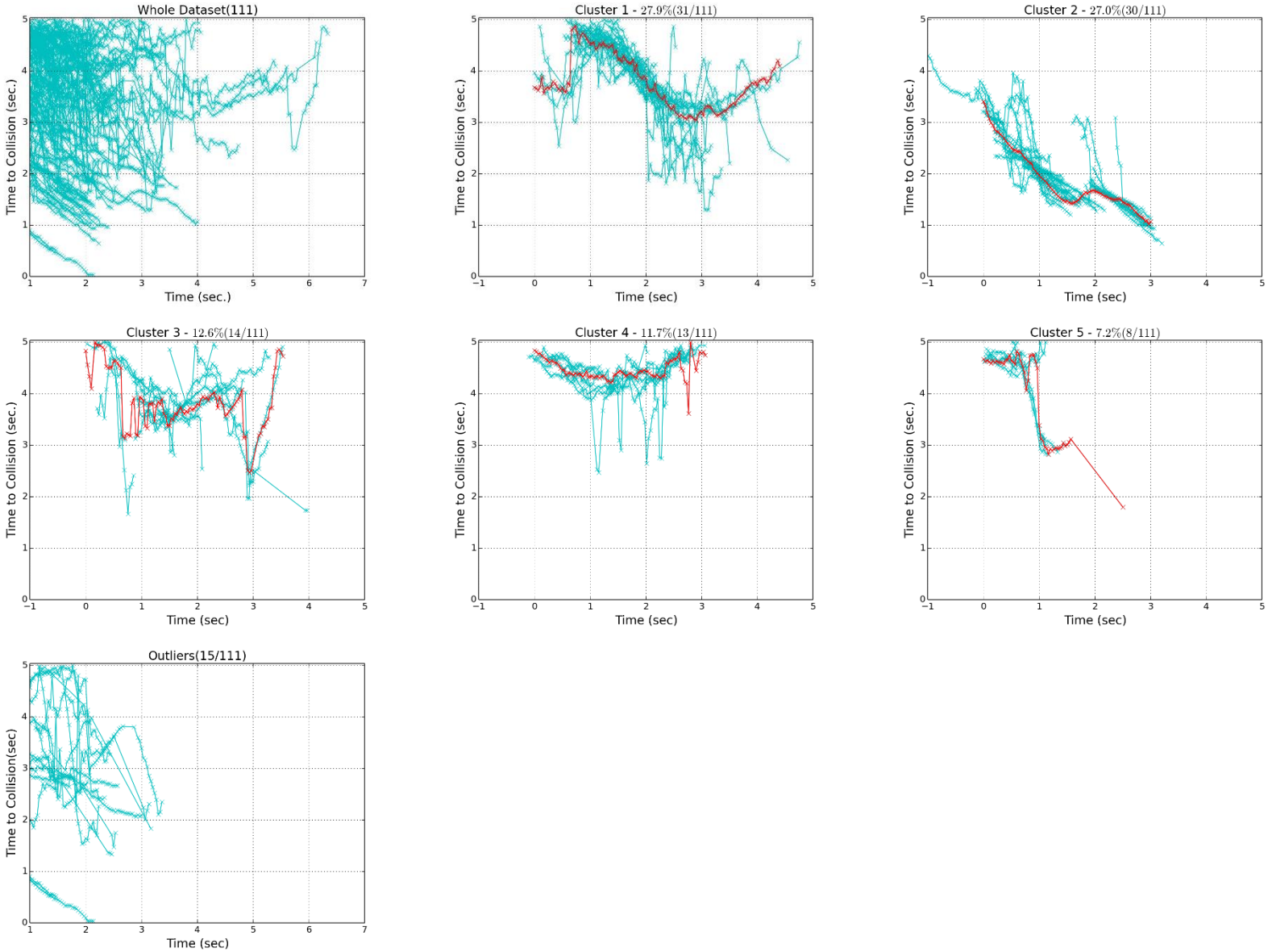


Figure 6.13: Clusters of the TTC profiles from the Guy Intersection (whole dataset, 5 clusters, and outliers)

6.7 Summary

The work presented in this chapter attempts to solve current deficiencies of surrogate safety analysis in terms of automatic computation and interpretation of several surrogate measures. The analysis is performed for the LTOD interactions which are the dominant collisions in a signalized intersection. First, the influence of the four types of road users' geometric representations on PET computation is evaluated. FBCH method was found as the most accurate method to compute PET. Second, four motion prediction methods are applied to predict future trajectories. The results show that the MPM prediction methods, especially MPSP model, produces the most robust indicators. This is particularly true for TTC, where the most dangerous values and the largest number of measurements were predicted using this method. An agreement between the interaction and collision frequencies can be verified using the proposed approach. Finally, two interpretation methods of safety indicators are investigated. The first method is based on analyzing the distribution of safety indicator values. The results show the superiority of motion pattern methods over the kinematic methods to evaluate the safety of interactions. The second analysis method is to go beyond current practices and research in surrogate safety analysis that relies on only one measurement of interaction indicators to assess the whole interaction. An important finding is that classifying indicators can eliminate the influence of false and noisy lower values, which could falsely classify an interaction as being less safe than is the case. These results help to understand the entire process of interactions with and without a collision and to identify which interaction may be used as a surrogate measure of safety.

CHAPTER 7 CONCLUSION AND RECOMMENDATIONS

7.1 Conclusion

The increasing availability of video data, through existing traffic cameras or dedicated field data collection, and the development of computer vision techniques pave the way for the collection of massive datasets about the microscopic behaviour of road users. This thesis was motivated by the need for an effective and generic framework to utilize and manage these rich amounts of spatio-temporal data for automated behaviour and safety analysis. The ultimate objective of this research is to propose a reliable tool able to diagnose safety deficiencies at any site without waiting for a collision to happen. The recorded video data, the main source of data for this approach, provides a detailed description of each road user and their interaction, and hence allows to better understand collision and interaction processes. This leads to extend a methodology to analyze interactions between road users and identify the encounters that could be reliably used as surrogate for collision. To achieve that, four main interrelated topics and their challenges were studied, and a consistent framework is proposed. It is composed of two main components: 1) a multi-level motion pattern learning sub-framework and 2) a surrogate safety analysis sub-framework. The former sub-framework is responsible to understand the behaviour and the basis of the latter sub-framework.

First of all, an improvement has been investigated to the existing video tracking tool to smooth extracted trajectories. The smoothing algorithm relies on extracted features of each road user to reconstruct a smoother road user trajectory, by estimating the relationship between the features and the object. The performance of the algorithm is evaluated on three real case studies visually and quantitatively. The algorithm performs well and can reduce noise based on the proposed quantitative criterion up to 97 % for vehicles.

Henceforth, a multi-level motion pattern learning sub-framework is introduced for automated scene interpretation and anomalous behaviour detection. One of the advantages of this sub-framework is to use actual trajectories as prototypes without any pre- or post-processing. The sub-framework is implemented in three different real cases which demonstrate the following advantages of the proposed sub-framework:

- It reduces computation up to 90 % for motion pattern learning,
- It reduces the required memory space up to 85 %,

- It connects incomplete trajectories with accuracy between 70 to 97 %,
- It removes the outliers before MP learning,
- It detects anomalous events such as abnormal left turn, improper turns, and excessive speed.

The sub-framework is tested in a traffic environment where it can be used to interpret any scene for surveillance purposes. Moreover, the improvements of computation cost in time and space pave the way to the use of the method in online systems.

The second component of the proposed framework and the ultimate goal of this research is to propose an automated methodology for surrogate safety analysis. Utilizing the output of the learning sub-framework, the learnt spatial and temporal prototypes are used to predict the intended behaviour trajectories and to compute several surrogate safety indicators such as TTC and pPET. The advantage of this method lies in its ability to implicitly take context into account such as the road geometry and traffic structures: this is exemplified when analyzing the distributions of collision points and crossing zones. To add further support, the safety analysis sub-framework includes kinematic methods such as constant velocity (the common prediction method used in most of surrogate safety methodologies) and normal adaptation. The kinematic methods predict collision points and conflict zones in areas that cannot be realistically expected to be reached in the studied interactions. This finding helps to conclude that kinematic methods are not realistic enough to compute safety indicators and to understand collision processes. Comparing the safety indicators based on MPM methods with the traditional motion prediction at CV on LTOD interactions, our MPM algorithms are able to compute the surrogate indicators early with a larger number of measurements. To sum up, the proposed framework based on MPM methods is able to perform the safety diagnosis of a signalized intersection in an automated and unsupervised fashion.

Secondly, a core requirement of surrogate safety analysis is to calculate reliable safety indicators. Hence, the geometric representation of the road user volume can affect the accuracy of the calculated indicators and consequently the safety evaluation of any site. Testing different volume representations on the calculation of PETs for LTOD interactions reveals that the FBCH method is highly correlated with the manual method and can compute PET with accuracy equals to 0.25 s when the camera coverage of the intersection is almost overhead. However, testing the methods for low angle FOV shows decreasing accuracy for estimating PET values. The important finding

is that the representation of road user volumes influences the accuracy of the computed PET and therefore the safety assessment of the studied interactions.

Finally, the research provides decision makers and safety analysts with two interpretation methods of continuous safety indicators to assess the safety of any site and/or interaction types. The first method is based on analyzing the distribution of an aggregated value of a safety indicator time series. The number of encounters, estimated using different motion prediction methods, was analyzed to quantify its correlation with the actual number of traffic collisions. This shows the superiority of the MPSP methods to evaluate the safety of studied interactions as it seems to be more consistent with the collision record. Moreover, this analysis proves the importance of choosing a reliable motion prediction method for the safety diagnosis of the studied site or for the comparison between different sites/interaction types. It also proves that the choice of a threshold is a very challenging task and this choice may affect the safety diagnosis as well.

A new interpretation method is proposed which aims to analyze the whole profile of safety indicators to assess the whole interaction. In this method, it is not required to identify a threshold for safety analysis. Instead, the clustering of indicators reveals typical profiles of indicators which can be applied to estimate and rank the safety of different interaction types. This method is applied to analyze the similarities between interactions with and without collisions which yields further credibility to the main hypothesis of surrogate safety analysis that some interactions without a collision have similar processes to collisions and could be used as predictors. It also strengthens the conclusion made in (Saunier, Mourji, & Agard, 2011) that not all interactions should be used for surrogate safety analysis. Furthermore, TTC time series for LTOD interactions are clustered and used to assess the safety of the different interaction types. An important finding is that clustering indicators is less sensitive to the false and noisy lower values, which could falsely classify an interaction as being less safe than is the case. In addition, it helps to understand the whole process of encounters.

Finally, this thesis is unique in the field of road safety analysis and transportation engineering in sharing the methods (the software code is released as open source (Mohamed M. G., 2014; Saunier N. , 2014)) to enable scientific reproducibility and encourage more collaboration in this area. It is believed that these tools can benefit other researchers and that the area of surrogate safety analysis,

with its many methods and indicators, can only progress if they can be compared by building upon each other's work.

7.2 Limitations

The limitations of the proposed framework are as follows:

- The selected parameters for the motion pattern learning method are the result of an iterative manual process of trial and error. This influences the prototypes number and needs to be carefully selected.
- Anomaly detection is dependent on tracking and grouping robustness which is highly affected by the camera FOV. Anomalies are manually validated and interpreted.
- The stopping zones need to be learnt with the POI model and the time spent in each zone should be learnt. This could be useful for activity learning and for motion prediction method.
- The measurement of safety indicators using geometric representation, in particular for models based on features, is less accurate when video is recorded with a low angle FOV.
- Some of the predicted trajectories extend out of the scene limit. Only potential collision points or crossing zones observed within the scene limits are considered in the safety analysis. It is recommended to limit the prediction within the scene limit because MPs cannot be learnt outside of the FOV and therefore only kinematic models can be used, which may not be realistic.

7.3 Perspectives and Future Works

Although this thesis provides several contributions for scene interpretation and surrogate safety analysis, there is still considerable room for further research.

- Improving computer vision techniques to extract more accurate trajectories, in particular for low angle FOV coverage: the location of the video camera has a significant influence on the quality of the extracted road user tracks using the TI tool and on the calculation of safety indicators.

- The analysis of large datasets will help the validation of the proposed methodology and its use to rank different sites.
- The framework need to be applied to include vulnerable road users (e.g. pedestrian and cyclist). It can be applied to diagnose the safety of vehicle- pedestrian interaction and vehicle-cyclist interactions.
- A reliable relationship between interactions without and with a collision should be established. This can be achieved by collecting interactions at different sites and having the associated collision data. This relationship can be used to estimate the expected collisions. Some researchers are working to find this relationship, but a reliable relationship is not constructed yet.
- Providing guidelines for using surrogate safety measures, in particular the use of indicators such as TTC and pPET.
- Interactions should also be clustered based on several indicators simultaneously.

REFERENCES

Allen, B. L., Shin, B. T., & Cooper, P. J. (1978). Analysis of Traffic Conflicts and Collisions. *Transportation Research Record 667*, 67-74.

Althoff, M., Stursberg, O., & Buss, M. (2008). Stochastic reachable sets of interacting traffic participants. *Intelligent Vehicles Symposium* (pp. 1086-1092). Eindhoven, The Netherlands: IEEE.

Archer, J. (2005). *Indicators for traffic safety assessment and prediction and their application in micro-simulation modelling:A study of urban and suburban intersections.* Stockholm: Sweden Royal Institute of Technology.

Atev, S., Arumugam, H., Masoud, O., Janardan, R., & Papanikolopoulos, N. P. (2005). A vision-based approach to collision prediction at traffic intersections. *Intelligent Transportation Systems, IEEE Transactions on, 6*(4), 416-423.

Atev, S., Masoud, O., & Papanikolopoulos, N. (2006, Oct). Learning Traffic Patterns at Intersections by Spectral Clustering of Motion Trajectories. *International Conference on Intelligent Robots and Systems* (pp. 4851-4856). IEEE/RSJ.

Bas, E., Tekalp, A. M., & Salman, F. S. (2007). Automatic vehicle counting from video for traffic flow analysis. *Intelligent Vehicles Symposium, 2007 IEEE,* (pp. 392-397).

Bennewitz, M., Burgard, W., Cielniak, G., & Thrun, S. (2005). Learning motion patterns of people for compliant robot motion. *Internationl Journal of Robotics Research, 24*, 31-48.

Berkhin, P. (2006). Survey Of Clustering Data Mining Techniques. In *Grouping multidimensional data* (pp. 25--71). Springer.

Bernardin, K., & Stiefelhagen, R. (2008). Evaluating multiple object tracking performance: the CLEAR MOT metrics. *Journal on Image and Video Processing, 2008*, 1.

Beymer, D., McLauchlan, P., Coifman, B., & Malik, J. (1997). A Real-time Computer Vision System for Measuring Traffic Parameters. *In Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)* (pp. 495-501). Washington, DC, USA: IEEE Computer Society.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *the Journal of machine Learning research, 3*, 993-1022.

Broadhurst, A. E., Baker, S., & Kanade, T. (2004). A prediction and planning framework for road safety analysis, obstacle avoidance and driver information. *In Proc. of the 11th World Congress on Intelligent Transportation Systems.*

Broadhurst, A. E., Baker, S., & Kanade, T. (2005). Monte Carlo Road Safety Reasoning. *IEEE Intelligent Vehicle Symposium*, (pp. 319-324). Las Vegas, NV.

Buch, N., Velastin, S., & Orwell, J. (2011, Sept). A Review of Computer Vision Techniques for the Analysis of Urban Traffic. *Intelligent Transportation Systems, IEEE Transactions on, 12*(3), 920-939.

Cunto, F. (2008). *Assessing Safety Performance of Transportation Systems using Microscopic Simulation.* Civil Engineering. Waterloo, Ontario, Canada: University of Waterloo.

Danielsson, S., Petersson, L., & Eidehall, A. (2007). Monte Carlo based Threat Assessment: Analysis and Improvements. *IEEE Intelligent Vehicles Symposium*, (pp. 233-238). Istanbul, Turkey.

Davis, G. A., Hourdos, J., Xiong, H., & Chatterjee, I. (2011). Outline for a causal model of traffic conflicts and crashes. *Accident Analysis and Prevention, 43*, 1907-1919.

Eidehall, A., & Petersson, L. (2006). Threat assessment for general road scenes using monte carlo sampling. *IEEE Intelligent Transportation Systems Conference*, (pp. 1173-1178). Toronto, Canada.

Eidehall, A., & Petersson, L. (2008). Statistical Threat Assessment for General Road Scenes Using Monte Carlo Sampling. *IEEE Transactions on Intelligent Transportation Systems, 9*(1), 137-147.

Elvik, R., Vaa, T., Erke, A., & Sorensen, M. (2009). *The Handbook of Road Safety Measures* (2nd Ed. ed.). Howard House, Wagon Lane, Bingley BD16 1WA, UK: Emerald Group Publishing Limited.

Ettehadieh, D., Farooq, B., & Saunier, N. (2015). Systematic parameter optimization and application of automated tracking in pedestrian-dominant situations. *Transportation Research Board Annual Meeting Compendium of Papers, paper #15-2400.*

Ferguson, T. S. (1973). A Bayesian analysis of some nonparametric problems. *The annals of statistics*, 209-230.

Fetters, L., & Todd, J. (1987). Quantitative Assessment of Infant Reaching Movements. *Journal of Motor Behavior, 19*(2), 147-166. Retrieved from http://dx.doi.org/10.1080/00222895.1987.10735405

Flash, T., & Hogans, N. (1985). The Coordination of Arm Movements: An Experimentally Confirmed Mathematical Model. *Journal of neuroscience, 5*, 1688-1703.

Fu, W., Wang, J., Li, Z., Lu, H., & Ma, S. (2012). Learning semantic motion patterns for dynamic scenes by improved sparse topical coding. *Multimedia and Expo (ICME), 2012 IEEE International Conference on*, (pp. 296-301).

Gettman, D., & Head, L. (2003). Surrogate safety measures from traffic simulation models. *Transportation Research Record: Journal of the Transportation Research Board, 1840*(1), 104-115.

Gettman, D., Pu, L., Sayed, T., & Shelby, S. G. (2008). *Surrogate safety assessment model and validation: Final report.* Tech. rep.

Green, E. R., Agent, K. R., & Pigman, J. G. (2005). *Evaluation of auto incident recording system (AIRS).* University of Kentucky: Research Report KRC-05-09, Kentucky Transportation Center.

Hansson, A. (1975). *Studies in driver behaviour, with applications in traffic design and planning. Two examples (in Swedish).* Lund: Lund Institute of Technology, Department of Traffic Planning and Engineering, University of Lund.

Hauer, E. (1982). Traffic Conflicts and Exposure. *Accident Analysis and Prevention, 14*, 359-364.

Hayward, J. C. (1972). Near-miss determination through use of a scale of danger. *Highway Research Record, 384*, 24-34.

Hofmann, T. (1999). Probabilistic latent semantic indexing. *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, (pp. 50-57).

Hogan, N., & Sternad, D. (2009). Sensitivity of smoothness measures to movement duration, amplitude, and arrests. *Journal of motor behavior, 41*(6), 529-534.

Hu, W., Tan, T., Wang, L., & Maybank, S. (2004, August). A Survey on Visual Surveillance of Object Motion and Behaviors. *IEEE Transactions on Systems, Man, and Cybernetics - Part C, 34*(3), 334-352.

Hu, W., Xiao, X., Fu, Z., Xie, D., Tan, T., & Maybank, S. (2006, Sept). A system for learning statistical motion patterns. *Pattern Analysis and Machine Intelligence, IEEE Transactions on, 28*(9), 1450-1464.

Hu, W., Xiao, X., Xie, D., & Tan, T. (2004). Traffic Accident Prediction Using 3-D Model-Based Vehicle Tracking. *IEEE Transactions on Vehicular Technology, 53*, pp. 677–694.

Hu, W., Xie, D., Fu, Z., Zeng, W., & Maybank, S. (2007). Semantic-based surveillance video retrieval. *Image Processing, IEEE Transactions on, 16*(4), 1168-1181.

Huang, P., Xu, Y., & Liang, B. (2006). Global minimum-jerk trajectory planning of space manipulator. *International Journal of Control Automation and Systems, 4*(4), 405.

Hydén. (1987). *The Development of a Method for Traffic Safety Evaluation: The Swedish Traffic Conflicts Technique.* Lund, Sweden: Dept. of Traffic Planning and Engineering, Lund University.

Hydén, C. (1996). Traffic Conflicts Technique: State-of-the-art. In H. Topp, *Traffic Safety Work with Video-Processing.* Kaiserslauten, Germany: University Kaiserslautern. Transportation Department, Green Series No.43,.

Hydén, C., & Amundsen, F. H. (1977). Proceedings: first workshop on traffic conflicts, Oslo 77. Norwegian Council for Scientific and Industrial Research.

Ismail, K. (2010). *Application of Computer Vision Techniques for Automated Road Safety Analysis and Traffic Data Collection.* Vancouver: University of British Columbia.

Jackson, S., Miranda-Moreno, L., St-Aubin, P., & Saunier, N. (2013). A Flexible, Mobile Video Camera System and Open Source Video Analysis Software for Road Safety and Behavioural Analysis. *Transportation Research Record: Journal of the Transportation Research Board, 2365*, 90-98.

Jodoin, J.-P., Bilodeau, G.-A., & Saunier, N. (2014). Urban Tracker: Multiple Object Tracking in Urban Mixed Traffic. *IEEE Winter Applications of Computer Vision Conference (WACV).*

Kahn, L., Zygman, M., Rymer, W., & Reinkensmeyer, D. (2001). Effect of robot-assisted and unassisted exercise on functional reaching in chronic hemiparesis. *Engineering in Medicine and Biology Society, 2001. Proceedings of the 23rd Annual International Conference of the IEEE, 2*, pp. 1344--1347.

Koller, D., Daniilidis, K., Thorhallson, T., & Nagel, H.-H. (1992). Model-based object tracking in traffic scenes. *Computer Vision—ECCV'92*, (pp. 437-452).

Lambert, A., Gruyer, D., Pierre, G., & Ndjeng, A. (2008). Collision Probability Assessment for Speed Control. *IEEE International Conference on Intelligent Transportation Systems*, (pp. 1043-1048). Beijing, China.

Laugier, C., Paromtchik, I., Perrollaz, M., Yong, M., Yoder, J., Tay, C., . . . Negre, A. (2011). Probabilistic Analysis of Dynamic Scenes and Collision Risks Assessment to Improve Driving Safety. *IEEE Intelligent Transportation Systems Magazine, 3*(4), 4-19.

Laureshyn, A. (2010). *Application of automated video analysis to road user behaviour*. Lund: Lund University.

Laureshyn, A., Åström, K., & Brundell-Freij, K. (2009). From Speed Profiles Data to Analysis of Behaviour: Classifictation by Pattern Recognition Technology. *IATSS Research 33 (2)*, (pp. 88-98).

Laureshyn, A., Svensson, Å., & Hydén, C. (2010, November). Evaluation of traffic safety,based on micro-level behavioural data: theoretical framework and first implementation. *Accident Analysis and Prevention, 42*(6), 1647-1646.

Liao, T. W. (2005). Clustering of time series data- a survey. *Pattern Recognition, 38*(11), 1857-1874.

Lipton, A., Fujiyoshi, H., & Patil, R. (1998, Oct). Moving target classification and tracking from real-time video. *Applications of Computer Vision, 1998. WACV '98. Proceedings., Fourth IEEE Workshop on*, (pp. 8-14).

Liu, M., Wu, C., & Zhang, Y. (2008). A Review of Traffic Visual Tracking Technology. *International Conference on Audio, Language and Image Processing (ICALIP)*, (pp. 1016-1020). Shanghai, China.

Maggio, E., & Cavallaro, A. (2011). *Video tracking : theory and practice.* Chichester, West sussex, UK: John Wiley & Sons, Ltd.

Makris, D., & Ellis, T. (2005, June). Learning semantic scene models from observing activity in visual surveillance. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 35*(3), 397-408.

Malik, J., & Russell, S. (1997). *Traffic Surveillance and Detection Technology Development: New Traffic Sensor Technology.* Berkeley: California PATH Research Final Report, University of California.

Malinovskiy, Y., Wu, Y.-J., & Wang, Y. (2009). Video-based vehicle detection and tracking using spatiotemporal maps. *Transportation Research Record: Journal of the Transportation Research Board, 2121*(1), 81-89.

McKenna, S., & Nait-Charif, H. (2004, Aug). Learning spatial context from tracking using penalised likelihoods. *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on, 4*, pp. 138-141 Vol.4.

Messelodi, S., Modena, C. M., & Zanin, M. (2005). A computer vision system for the detection and classification of vehicles at urban road intersections. *Pattern analysis and applications, 8*(1-2), 17-31.

Minderhoud, M. M., & Bovy, P. H. (2001). Extended time-to-collision measures for road traffic safety assessment. *Accident Analysis and Prevention, 33*, 89-97.

Mohamed, M. G. (2014). *https://bitbucket.org/mohamedgomaa/behaviour-analysis*. Retrieved from Behaviour Analysis.

Mohamed, M. G., & Saunier, N. (2013a). Motion prediction methods for surrogate safety analysis. *Transportation Research Record: Journal of the Transportation Research Board, 2386*(1), 168-178.

Mohamed, M. G., & Saunier, N. (2013b, #may#). Classifying Profiles of Surrogate Safety Measures to Understand Collision Processes. *Canadian Multidisciplinary Road Safety Conference.* Montreal.

Mohamed, M. G., & Saunier, N. (2013c, #may#). The Influence of Motion Prediction Methods on Surrogate Safety Measures. *CSCE Annual Conference.* Montreal.

Mohamed, M. G., Saunier, N., Miranda-Moreno, L., & Ukkusuri, S. V. (2013). A clustering regression approach: A comprehensive injury severity analysis of pedestrian-vehicle crashes in New York, US and Montreal, Canada. *Safety Science, 54*, 27-37.

Morris, B. T., & Trivedi, M. M. (2008b). Learning, modeling, and classification of vehicle track patterns from live video. *Intelligent Transportation Systems, IEEE Transactions on, 9*(3), 425-437.

Morris, B. T., & Trivedi, M. M. (2013). Understanding vehicular traffic behavior from video: a survey of unsupervised approaches. *Journal of Electronic Imaging, 22*(4), (pp41113-1:41113-15).

Morris, B., & Trivedi, M. (2008a). A Suvey of Vision-Based Trajectory Learning and Analyis for Surveillance. *IEEE Transactions on Circuits and Systems for Video Technology*, *18*, pp. 1114-1127.

Morris, B., & Trivedi, M. (2009). Learning Trajectory Patterns by Clustering: Experimental Studies and Comparative Evaluation. *Computer Vision and Pattern Recognition* (pp. 312- 319). IEEE Computer Society Conference on.

Morris, B., & Trivedi, M. (2011). Trajectory Learning for Activity Understanding: Unsupervised, Multilevel, and Long-Term Adaptive Approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 33*(11), 2287-2301.

Najm, W. G., Smith, J. D., & Smith, D. L. (2001). *Analysis of crossing path crashes.* Tech. rep.

Nedrich, M., & Davis, J. (2013). Detecting behavioral zones in local and global camera views. *Machine Vision and Applications, 24*(3), 579-605. Retrieved from http://dx.doi.org/10.1007/s00138-012-0418-4

Ng, A. Y., Jordan, M. I., & Weiss, Y. (2002). On Spectral Clustering: Analysis and an algorithm. *Advances in neural information processing systems* (pp. 849-856). MIT Press.

Nicholson, A. (1985). The variability of accident counts. *Accident Analysis & Prevention, 17*(1), 47-56.

Parkhurst, D. (2006). *Using digital video analysis to monitor driver behaviour at intersections.* Center for Transportation Research and Education, Iowa State University.

Pearl, J. (2000). *Causality: Models, Reasoning, and Inference.* Cambridge, UK: Cambridge University Press.

Piciarelli, C., & Foresti, G. L. (2006). On-line trajectory clustering for anomalous events detection. *Pattern Recognit. Lett*, 1835-1842.

Ragland, D. R., & Zabyshny, A. A. (2003). Intersection decision support project: Taxonomy of crossing-path crashes at intersections using GES 2000 data. *Safe Transportation Research & Education Center*.

Rohrer, B., Fasoli, S., Krebs, H. I., Hughes, R., Volpe, B., Frontera, W. R., . . . Hogan, N. (2002). Movement smoothness changes during stroke recovery. *The Journal of Neuroscience, 22*(18), 8297-8304.

RSM. (2003). *Road Safety Manual.* La Grande Arche: PIARC Technical Committee on Road Safety - World Road Safety.

SAAQ. (2015). *Bilan routier.* Quebec: Société de l'assurance automobile du Québec (SAAQ). Retrieved May 11, 2015, from http://www.saaq.gouv.qc.ca/securite_routiere/bilan_routier_2014/index.php

Saleemi, I., Shafique, K., & Shah, M. (2009). Probabilistic modeling of scene dynamics for applications in visual surveillance. *Pattern Analysis and Machine Intelligence, IEEE Transactions on, 31*(8), 1472-1485.

Saunier, N. (2014). *Traffic Intelligence software*. Retrieved 12 26, 2014, from https://bitbucket.org/Nicolas/trafficintelligence

Saunier, N., & Mohamed, M. G. (2014). Clustering Surrogate Safety Indicators to Understand Collision Processes. *Transportation Research Board, 93rd meeting.* Washington,USA.

Saunier, N., & Sayed, T. (2006). A feature-based tracking algorithm for vehicles in intersections. *The 3rd Canadian Conference on Computer and Robot Vision.* IEEE.

Saunier, N., & Sayed, T. (2007). Automated Road Safety Analysis Using Video Data. *Transportation Research Record: Journal of the Transportation Research Board, 2019*, 57-64.

Saunier, N., & Sayed, T. (2008). Probabilistic Framework for Automated Analysis of Exposure to Road Collisions. *Transportation Research Record: Journal of the Transportation Research Board, 2083*, 96-104.

Saunier, N., Mourji, N., & Agard, B. (2011). Mining microscopic data of vehicle conflicts and collisions to investigate collision factors. *Transportation Research Record: Journal of the Transportation Research Board, 2237*, 41-50.

Saunier, N., Sayed, T., & Ismail, K. (2010). Large Scale Automated Analysis of Vehicle Interactions and Collisions. *Transportation Research Record: Journal of the Transportation Research Board, 2147*, 42-50.

Saunier, N., Sayed, T., & Ismail, K. (2010). Large Scale Automated Analysis of Vehicle Interactions and Collisions. *Transportation Research Record: Journal of the Transportation Research Board, 2147*, 42-50.

Saunier, N., Sayed, T., & Lim, C. (2007, #oct#). Probabilistic Collision Prediction for Vision-Based Automated Road Safety Analysis. *The 10th International IEEE Conference on Intelligent Transportation Systems* (pp. 872-878). Seattle: IEEE.

Savitzky, A., & Golay, M. J. (1964). Smoothing and differentiation of data by simplified least squares procedures. *Analytical chemistry, 36*(8), 1627-1639.

Sekiyama, N., Minoura, K., & Watanabe, T. (2011). Prediction of collisions between vehicles using attainable region. *International Conference on Ubiquitous Information Management and Communication (ICUIMC)*, (pp. 34:1--34:6). Seoul, Korea.

Sobral, A. (2013, Jun). {BGSLibrary}: An OpenCV C++ Background Subtraction Library. *IX Workshop de Visão Computacional (WVC'2013)*. Rio de Janeiro, Brazil. Retrieved from https://github.com/andrewssobral/bgslibrary

Sobral, A., & Vacavant, A. (2014, #may#). A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. *Computer Vision and Image Understanding, 122*(0), 4-21. Retrieved from http://www.sciencedirect.com/science/article/pii/S1077314213002361

Song, L., Jiang, F., Shi, Z., Molina, R., & Katsaggelos, A. K. (2014). Toward Dynamic Scene Understanding by Hierarchical Motion Pattern Mining. *IEEE Transactions on Intelligent Transportation Systems*, 1273--1285.

Songchitruksa, P., & Tarko, A. (2004). *Using Imaging Technology to Evaluate Highway Safety*. Purdue University.

Songchitruksa, P., & Tarko, A. P. (2006). The extreme value theory approach to safety estimation . *Accident Analysis & Prevention , 38*(4), 811-822. Retrieved from http://www.sciencedirect.com/science/article/pii/S0001457506000236

Sorstedt, J., Svensson, L., Sandblom, F., & Hammarstrand, L. (2011). A New Vehicle Motion Model for Improved Predictions and Situation Assessment. *Intelligent Transportation Systems, IEEE Transactions on, 12*(4), 1209-1219.

Spagnolo, P., Leo, M., D'Orazio, T., & Distante, A. (2004). Robust moving objects segmentation by background subtraction. *The International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS).*

St-Aubin, P., Saunier, N., & Miranda-Moreno, L. (2015). Large-scale automated proactive road safety analysis using video data. *Transportation Research Part C: Emerging Technologies*. Retrieved from http://www.sciencedirect.com/science/article/pii/S0968090X15001485

Stauffer, C. (2003, June). Estimating Tracking Sources and Sinks. *Computer Vision and Pattern Recognition Workshop, 2003. CVPRW '03. Conference on, 4*, pp. 35-35.

Svensson, Å. (1998). *A method for analysing the traffic process in a safety perspective*. Lund: Lund Institute of Technology.

Svensson, A., & Hydén, C. (2006). Estimating the severity of safety related behaviour. *Accident Analysis & Prevention, 38*(2), 379-385.

Tarko, A., Davis, G. A., Saunier, N., Sayed, T., & Washington, S. (2009). *Surrogate Measures of Safety.* White Paper, ANB20(3) Subcommittee on Surrogate Measures of Safety.

Teh, Y. W., Jordan, M. I., Beal, M. J., & Blei, D. M. (2006). Hierarchical dirichlet processes. *Journal of the american statistical association, 101*(476).

Thrun, S., Burgard, W., & Fox, D. (2005). *Probabilistic robotics.* MIT Press.

Tomasi, C., & Kanade, T. (1991). *Detection and tracking of point features.* School of Computer Science, Carnegie Mellon Univ. Pittsburgh.

Vlachos, M., Kollios, G., & Gunopulos, D. (2002). Discovering similar multidimensional trajectories. *Data Engineering, Proceedings. 18th International Conference on IEEE* (pp. 673-684). IEEE.

Vodden, K., Smith, D., Eaton, F., & Mayhew, D. (2007). *Analysis and Estimation of the Social Cost of Motor Vehicle Collisions in Ontario.* Transport Canada.

Wang, X. (2011). Action recognition using topic models. In *Visual Analysis of Humans* (pp. 311-332). Springer.

Wang, X., Ma, K. T., Ng, G.-W., & Grimson, W. E. (2011). Trajectory analysis and semantic region modeling using nonparametric hierarchical bayesian models. *International journal of computer vision, 95*(3), 287-312.

Wang, X., Ma, X., & Grimson, W. E. (2009). Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on , 31*(3), 539-555.

Wang, X., Tieu, K., & Grimson, E. (2006). Learning Semantic Scene Models by Trajectory Analysis. *In ECCV (3)*, (pp. 110-123).

Wen, J., Li, C., & Xiong, Z. (2011). Behavior pattern extraction by trajectory analysis. *Frontiers of Computer Science in China, 5*, 37-44. Retrieved from http://dx.doi.org/10.1007/s11704-010-0074-7

WHO. (2009). *Global status report on road safety: time for action.* Geneva: World Health Organization.

WHO. (2013). *Global status report on road safety 2013: supporting a decade of action.* World Health Organization.

Xu, R., & Wunsch, D. (2005, #may#). Survey of clustering algorithms. *Neural Networks, IEEE Transactions on, 16*(3), 645-678. Retrieved from http://dx.doi.org/10.1109/tnn.2005.845141

Yi, Z., & Liangzhong, F. (2010). Moving object detection based on running average background and temporal difference. *Intelligent Systems and Knowledge Engineering (ISKE), 2010 International Conference on*, (pp. 270-272).

Zaki, M. H., Sayed, T., Ismail, K., & Alrukaibi, F. (2012). Use of computer vision to identify pedestrians' nonconforming behavior at urban intersections. *Transportation Research Record: Journal of the Transportation Research Board, 2279*(1), 54-64.

Zhao, R., & Wang, X. (2013). Counting Vehicles from Semantic Regions. *Intelligent Transportation Systems, IEEE Transactions on, 14*(2), 1016-1022.

Zheng, L., Ismail, K., & Meng, X. (2014). Shifted Gamma-Generalized Pareto Distribution model to map the safety continuum and estimate crashes. *Safety Science, 64*, 155-162.

Zhu, Q. (1990). A stochastic algorithm for obstacle motion prediction in visual guidance of robot motion. *IEEE International Conference on Systems Engineering*, (pp. 216-219).

APPENDIX A – POI DETECTION ALGORITHMS

A.1 Gaussian Mixture Model (GMM)

Let x is a d-dimensional data points, the GMM is a probabilistic model that assume that x is generated from a mixture of Gaussian distribution. A single multivariate Gaussian model is defined by the probabilistic density function (pdf) in the following equation:

$$p(\,x|\mu,\sum\,) = \frac{e^{-\frac{1}{2}(x-\mu)^T\sum^{-1}(x-\mu)}}{(2\pi)^{\frac{d}{2}}.|\sum|^{\frac{1}{2}}}$$

Where $\mu,\sum$ are respectively the mean and of the covariance matrix of the distribution.

The GMM is a weighted sum of a finite number (K) of Gaussian distribution as given by the following equation for the pdf,

$$p(x|\theta) = \sum_{i=1}^{K} w_i N(\,x_i|\mu_i,\sigma_i)$$

Where: w_i is the weight (prior probability) of each component (and the sum of weights equals 1), θ is the set of all parameters $(w,\mu,\sum)$ for each distribution.

The GMM is visualized by a set of hyper-ellipsoids. Each hyper-ellipsoid is drawn by its center (μ) with an orientation equal to the eigenvectors of $\sum$. In our case, the points are two-dimensional and the GMM is therefore visualized as set of 2D ellipses.

A.2 Expectation-Maximisation (EM) Algorithm

The EM algorithm is an iterative optimization method to estimate a set of parameters θ, given an observed dataset X =$\{x_1,..,x_N\}$, where N is the number of observations in the sample. The objective of the EM algorithm is to adapt the unknown θ to maximize the likelihood L(θ|x) which is determined by the marginal likelihood of X as follows:

$$L(\theta|X) = \; p(X|\theta) = \sum_{i=1}^{N} p(x_i|\theta)$$

First, the unknown parameters are initialized $\theta_{t=0}$, then two steps are iteratively applied in the algorithm till the convergence.

Expectation step (E step): it estimates the expected value of the log likelihood function given X and the estimated parameters from the previous iteration θ_t.

$$Q(\theta|\theta_t) = E_{X|\theta_t}[\log L(\theta|X)]$$

Maximization step (M step): it estimates the parameters that maximize the E step

$$\theta_{t+1} = argmax_\theta Q(\theta|\theta_t)$$

APPENDIX B – PUBLISHED PAPERS

1. **Mohamed, M. G.**, & Saunier, N. (2015). "*Behaviour Analysis using a Multi-Level Motion Pattern Learning Framework*." Transportation Research Record: Journal of the Transportation Research Board, No. 2528, 116-127. DOI: 10.3141/2528-13.
2. Saunier, N. & **Mohamed, M. G.**, (2014). "*Clustering surrogate safety indicators to understand collision processes*." In Transportation Research Board Annual Meeting Compendium of Papers, 2014. 14-2380.
3. **Mohamed, M. G.**, & Saunier, N. (2013). "*Motion Prediction Methods for Surrogate Safety Analysis.*" Transportation Research Record: Journal of the Transportation Research Board, No. 2386, 168-178. DOI: 10.3141/2386-19.
4. **Mohamed, M. G.**, Saunier, N., Miranda-Moreno, L. F., & Ukkusuri, S. V. (2013). "*A clustering regression approach: A comprehensive injury severity analysis of pedestrian-vehicle crashes in New York, US} and Montreal, Canada."* Safety Science , 54(0), 27-37. Presented in Transportation Research Board (TRB) 92nd Annual Meeting, Washington, D.C., January 2012.DOI: 10.1016/j.ssci.2012.11.001.
5. **Mohamed, M. G.**, & Saunier, N. (2013). "*Classifying profiles of surrogate safety measures to understand collision processes*." In Canadian Multidisciplinary Road Safety Conference, Montreal, May 2013.
6. **Mohamed, M. G.**, & Saunier, N. (2013). "*The influence of motion prediction methods on surrogate safety measures*". In CSCE Annual Conference, Montreal, May 2013.